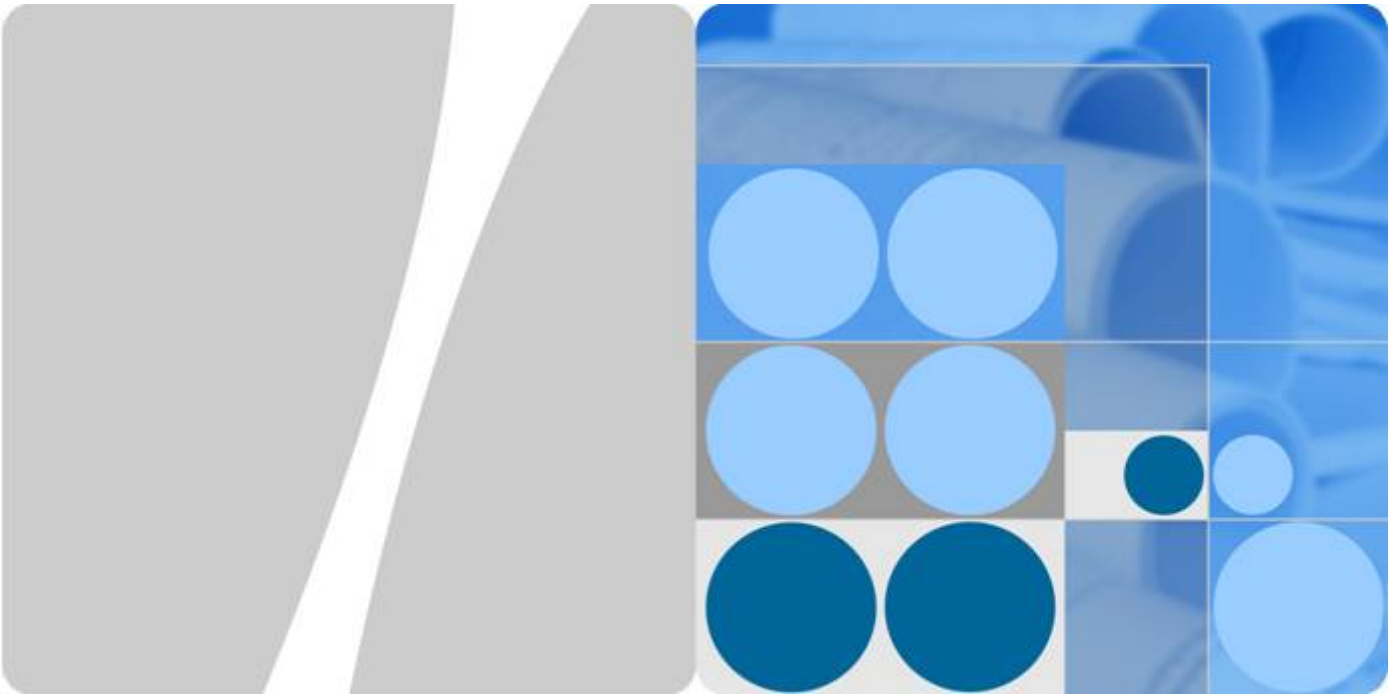


MapReduce Service

User Guide

Date **2025-02-07**

Contents

1 Overview

1.1 What Is MRS?

Big data is a huge challenge facing the Internet era as the data volume and types increase rapidly. Conventional data processing technologies, such as single-node storage and relational databases, are unable to solve the emerging big data problems. In this case, the Apache Software Foundation (ASF) has launched an open source Hadoop big data processing solution. Hadoop is an open source distributed computing platform that can fully utilize computing and storage capabilities of clusters to process massive amounts of data. If enterprises deploy Hadoop systems by themselves, the disadvantages include high costs, long deployment period, difficult maintenance, and inflexible use.

To solve the preceding problems, the cloud provides MapReduce Service (MRS) for managing the Hadoop system. With MRS, you can deploy a Hadoop cluster in just one click. MRS provides enterprise-level big data clusters on the cloud. Tenants can fully control clusters and easily run big data components such as Storm, Hadoop, Spark, HBase, and Kafka. MRS is fully compatible with open source APIs, and incorporates advantages of the cloud computing and storage and big data industry experience to provide customers with a full-stack big data platform featuring high performance, low cost, flexibility, and ease-of-use. In addition, the platform can be customized based on service requirements to help enterprises quickly build a massive data processing system and discover new value points and business opportunities by analyzing and mining massive amounts of data in real time or in non-real time.

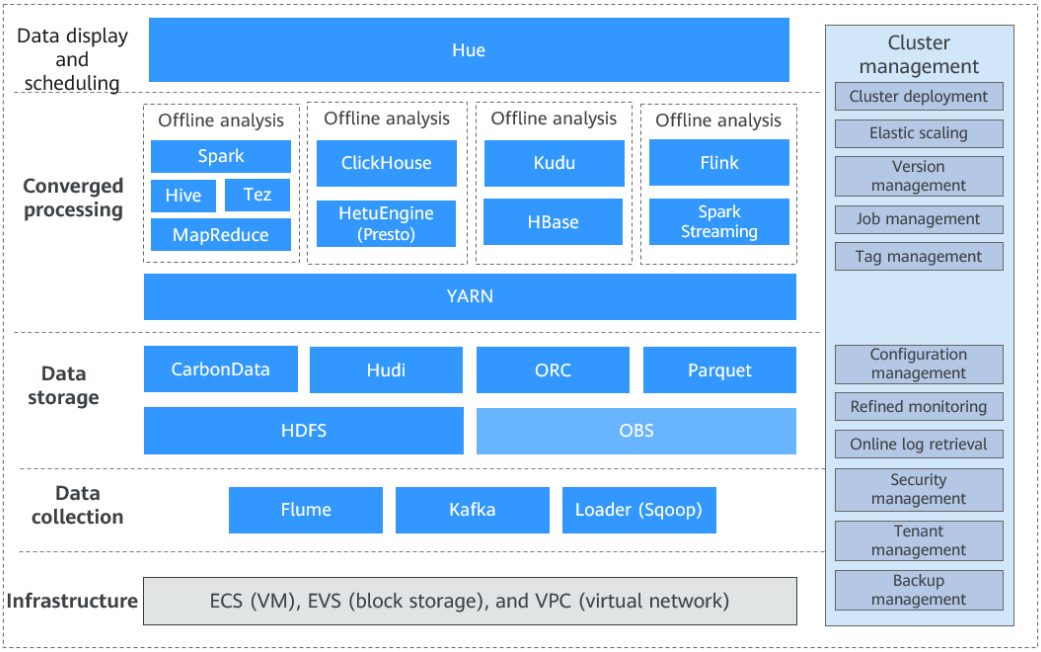
Product Architecture

Figure 1-1 shows the MRS logical architecture.

NOTE

MRS 3.x or later does not support patch management on the management console.

Figure 1-1 MRS architecture



MRS architecture includes infrastructure and big data processing phases.

- Infrastructure

  MRS big data clusters are built based on Elastic Cloud Server (ECS), and make full use of the high reliability and security capabilities of the virtualization layer.

  - A Virtual Private Cloud (VPC) is a virtual internal network provided for each tenant. It is isolated from other networks by default.
  - Elastic Volume Service (EVS) provides highly reliable and high-performance storage.
  - ECS provides scalable VMs, and works with VPCs, security groups, and the EVS multi-replica mechanism to build an efficient, reliable, and secure computing environment.

- Data integration

  The data integration layer provides data access capabilities of MRS clusters, including components Flume (data ingestion), Loader (relational data import), and Kafka (highly reliable message queue). Data can be imported to MRS clusters from various data sources.

- Data storage

  MRS clusters can store structured and unstructured data, and support multiple efficient formats to meet the requirements of different computing engines.

  - HDFS is a general-purpose distributed file system on a big data platform.
  - OBS is an object storage service that features high availability and low cost.
  - HBase supports data storage with indexes, and is applicable to high-performance index-based query scenarios.

- Data computing

  MRS provides multiple mainstream computing engines, including Storm (stream computing), MapReduce (batch processing), Tez (DAG model), Spark (in-memory computing), SparkStreaming (micro-batch stream computing), and Flink (stream computing), to meet the requirements of various big data application scenarios. The engines convert data structures and logic into data models that meet service requirements.

- Data analysis

  Based on the preset data model and easy-to-use SQL data analysis, users can select Hive (data warehouse), SparkSQL, and Presto (interactive query engine).

- Data display and scheduling

  To present data analysis results, MRS is integrated with Data Lake Factory (DLF), which is a one-stop big data collaboration development platform, to help you easily complete multiple tasks, such as data modeling, data integration, script development, job scheduling, and job monitoring. This makes big data more accessible than ever before, helping you quickly build big data processing centers.

- Cluster management

  All components of the Hadoop-based big data ecosystem are deployed in distributed mode, and their deployment, management, and O&M are complex.

  MRS provides a unified O&M management platform for cluster management, supporting one-click cluster deployment, multi-version selection, as well as manual scaling and auto scaling of clusters without service interruption. In addition, MRS provides job management, resource tag management, and O&M of the preceding data processing components at each layer. It also provides one-stop O&M capabilities, covering monitoring, alarm reporting, configuration, and patch upgrade.

Product Advantages

MRS has a powerful Hadoop kernel team and is deployed based on enterprise-level FusionInsight big data platform. MRS has been deployed on tens of thousands of nodes and can ensure Service Level Agreements (SLAs) for multi-level users.

MRS has the following advantages:

- High performance

  MRS supports self-developed CarbonData storage technology. CarbonData is a high-performance big data storage solution. It allows one data set to apply to multiple scenarios and supports features, such as multi-level indexing, dictionary encoding, pre-aggregation, dynamic partitioning, and quasi-real-time data query. This improves I/O scanning and computing performance and returns analysis results of tens of billions of data records in seconds. In addition, MRS supports self-developed enhanced scheduler Superior, which breaks the scale bottleneck of a single cluster and is capable of scheduling over 10,000 nodes in a cluster.

- Cost-effectiveness

Based on diversified cloud infrastructure, MRS provides various computing and storage choices and separates computing from storage, delivering cost-effective massive data storage solutions. MRS supports auto scaling to address peak and off-peak service loads, releasing idle resources on the big data platform for customers. MRS clusters can be created and scaled out when you need them, and can be terminated or scaled in after you use them, minimizing cost.

- High security

  MRS delivers enterprise-level big data multi-tenant permissions management and security management to support table-based and column-based access control and data encryption.

- Easy O&M

  MRS provides a visualized big data cluster management platform, improving O&M efficiency. MRS supports rolling patch upgrade and provides visualized patch release information and one-click patch installation without manual intervention, ensuring long-term stability of user clusters.

- High reliability

  The proven large-scale reliability and long-term stability of MRS meet enterprise-level high reliability requirements. In addition, MRS supports automatic data backup across AZs and regions, as well as automatic anti-affinity. It allows VMs to be distributed on different physical machines.

1.2 Advantages of MRS Compared with Self-Built Hadoop

MRS provides enterprise-level big data clusters on the cloud. Tenants can fully control the clusters and run big data components such as Hadoop, Spark, HBase, Kafka, and Storm with ease. MRS frees you from hardware purchase and maintenance. MRS is built based on FusionInsight big data enterprise-class platform, and has been deployed on tens of thousands of nodes in the industry, providing multi-level SLA assurance with professional Hadoop kernel service support. Compared with self-built Hadoop clusters, MRS has the following advantages:

1. **MRS supports one-click cluster creation, deletion, and scaling. You can use an elastic IP address (EIP) to access MRS Manager, making big data clusters easier to use.**
   - Self-built big data clusters pose problems such as high costs, long periods, difficult and inflexible O&M. To solve these problems, MRS provides one-click cluster creation, deletion, scale-out, and scale-in, allowing you to customize the cluster type, component range, number of nodes of each type, VM specifications, availability zones (AZs), VPC network, and authentication information. MRS can automatically create a cluster that meets the configuration requirements. In addition, you can quickly create multi-application clusters, for example, Hadoop analysis cluster, HBase cluster, and Kafka cluster. MRS supports heterogeneous cluster deployment. That is, VMs of different specifications can be combined in a cluster based on CPU types, disk capacities, disk types, and memory sizes.

- MRS provides an EIP-based secure channel for you to easily access the web UIs of components. This is more convenient than binding an EIP by yourself, and you can access the web UIs with a few clicks, avoiding the steps for logging in to a VPC, adding security group rules, and obtaining a public IP address.
- MRS provides custom bootstrap actions to flexibly configure your dedicated clusters. Third-party software that is not supported by MRS can be automatically installed, allowing you to perform custom operations such as modifying the cluster running environment.
- MRS supports the WrapperFS feature, provides the OBS translation capability (that is, access to OBS through address mapping) and can smoothly migrate data from HDFS to OBS. After migration, you can access the data stored in OBS from clients without modifying service code logic.

2. **MRS supports auto scaling, which is more cost-effective than the self-built Hadoop cluster.**

   MRS supports auto scaling to address peak and off-peak service loads. It applies for extra resources during peak hours and releases idle resources during off-peak hours, helping you save idle resources on the big data platform during off-peak hours, minimize costs, and focus on core services.

   In big data applications, especially in periodic data analysis and processing, cluster computing resources need to be dynamically adjusted based on service data changes to meet service requirements. The auto scaling function of MRS enables clusters to be elastically scaled out or in based on cluster loads. In addition, if the data volume changes regularly and you want to scale out or in a cluster before the data volume changes, you can use the MRS resource plan feature. MRS supports two types of auto scaling policies: auto scaling rules and resource plans

   - Auto scaling rules: You can increase or decrease Task nodes based on real-time cluster loads. Auto scaling will be triggered when the data volume changes but there may be some delay.
   - Resource plans: If the data volume changes periodically, you can create resource plans to resize the cluster before the data volume changes, thereby avoiding a delay in increasing or decreasing resources.

   Both auto scaling rules and resource plans can trigger auto scaling. You can configure both of them or configure one of them. Configuring both resource plans and auto scaling rules improves the cluster node scalability to cope with occasionally unexpected data volume peaks.

3. **MRS supports storage-compute decoupling, greatly improving the resource utilization of big data clusters.**

   In the traditional big data architecture where storage and compute resources are integrated, scaling-out is difficult and resources are not well-utilized. To solve these problems, MRS adopts a compute-storage separation architecture. Based on OBS, the storage achieves 99.999999999% reliability and unlimited capacity, supporting continuous growth of enterprise data. Computing resources can be elastically scaled in or out from 0 to *N* nodes. Hundreds of nodes can be quickly provisioned. With the new architecture, compute nodes can be elastically scaled. OBS-based cross-AZ data storage ensures higher reliability, frees you from worrying about emergencies such as earthquakes and fiber cuts. Storage and compute resources can be flexibly configured and

elastically scaled as required. This makes resource allocation more accurate and reasonable, greatly improving the resource utilization of big data clusters and reducing the comprehensive analysis cost by 50%.

In addition, the high performance compute-storage separation architecture breaks the limit of parallel computing of the integrated storage-compute architecture. It maximizes the high bandwidth and high concurrency of OBS, and optimizes the data access efficiency and in-depth parallel computing (such as metadata operation and write algorithm optimization) to improve higher performance.

4. **MRS supports self-developed CarbonData and Superior Scheduler, delivering better performance.**
   - MRS supports self-developed CarbonData storage technology. CarbonData is a high-performance big data storage solution. It allows one data set to apply to multiple scenarios and supports features, such as multi-level indexing, dictionary encoding, pre-aggregation, dynamic partitioning, and quasi-real-time data query. This improves I/O scanning and computing performance and returns analysis results of tens of billions of data records in seconds.
   - In addition, MRS supports self-developed Superior Scheduler, which enhances the scaling capability of a single cluster and is capable of scheduling over 10,000 nodes in a cluster. Superior Scheduler is a scheduling engine designed for the Hadoop YARN distributed resource management system. It is a high-performance and enterprise-level scheduler designed for converged resource pools and multi-tenant service requirements. Superior Scheduler achieves all functions of open-source schedulers, Fair Scheduler, and Capacity Scheduler. Compared with the open-source schedulers, Superior Scheduler is enhanced in the enterprise multi-tenant resource scheduling policy, resource isolation and sharing by multiple users in a tenant, scheduling performance, system resource utilization, and cluster scalability, and is designed to replace open source schedulers.
5. **MRS optimizes software and hardware based on Kunpeng processors to fully release hardware computing power and achieve cost-effectiveness.**

   MRS supports self-developed Kunpeng servers whose multi-core and high-concurrency capabilities are fully utilized to provide full-stack self-optimized chips, and uses self-developed EulerOS, JDK, and data acceleration layer to ensure hardware performance, delivering high computing power for big data computing. With the similar performance, the cost of the end-to-end big data solution is reduced by 30%.
6. **MRS supports multiple isolation modes and multi-tenant permission management of enterprise-level big data, ensuring higher security.**
   - MRS supports resource deployment and isolation of physical resources in dedicated zones. You can flexibly combine computing and storage resources, such as dedicated computing resources + shared storage resources, shared computing resources + dedicated storage resources, and dedicated computing resources + dedicated storage resources. An MRS cluster supports multiple logical tenants. Permission isolation enables the computing, storage, and table resources of the cluster to be divided based on tenants.
   - With Kerberos authentication, MRS provides role-based access control (RBAC) and sound audit functions.

- With Cloud Trace Service (CTS) being interconnected with MRS, you are provided with operation records of MRS resource operation requests and request results for querying, auditing, and backtracking. You can use CTS to audit and trace all cluster operations.
- It is proved that with Host Security Service (HSS) interconnected with MRS, service security is enhanced without deteriorating functions and performance.
- MRS supports unified user login based on web UI. Manager provides user authentication, which grants you permission to access a cluster.
- MRS supports data storage encryption, encrypted storage of all user accounts and passwords, encrypted transmission of data channels, and bidirectional certificate authentication for cross-trusted-zone data access of service modules.
- MRS big data clusters provide a complete multi-tenant solution for enterprise-level big data. Multi-tenant refers to a collection of multiple resources (each resource set is a tenant) in an MRS big data cluster. It can allocate and schedule resources, including computing and storage resources. Multi-tenant isolates the resources of a big data cluster into resource sets. Users can lease desired resource sets to run applications and jobs and store data. In a big data cluster, multiple resource sets can be deployed to meet diverse requirements of multiple users.
- MRS supports fine-grained permission management. With the fine-grained authorization capability provided by CLOUD IAM, MRS can specify the operations, resources, and request conditions of specific services. This mechanism allows for more flexible policy-based authorization, meeting requirements for secure access control. For example, you can grant MRS users only the permissions for performing specified operations on MRS clusters, such as creating a cluster and querying a cluster list rather than deleting a cluster. In addition, MRS supports fine-grained permission management of OBS for multiple tenants. Permissions to access OBS buckets and objects in the buckets are differentiated based on user roles, so that MRS users can each control a different directory in OBS buckets.
- MRS supports enterprise project management. The enterprise project is one way of managing cloud resources. Enterprise Management provides comprehensive management services for enterprise customers, such as cloud resources, personnel, permissions, and financial statuses. Common management consoles are oriented to the control and configuration of individual cloud products. The Enterprise Management console, in contrast, is more focused on resource management. It is designed to help enterprises manage cloud-based resources, personnel, permissions, and finances, in a hierarchical management manner, such as management of companies, departments, and projects. MRS allows users who have enabled Enterprise Project Management Service (EPS) to configure enterprise projects for a cluster during cluster creation and use EPS to manage MRS resources by group. This feature is applicable to scenarios where you need to manage multiple resources by group and perform operations such as permission control and project-based fee query on enterprise projects.

7. **MRS implements HA for all management nodes and supports comprehensive reliability mechanism, making the system more reliable.**

Based on Apache Hadoop open-source software, MRS optimizes and improves the reliability of main service components.

- HA for all management nodes

  In the Hadoop open-source version, data and compute nodes are managed in a distributed system, in which a single point of failure (SPOF) does not affect the operation of the entire system. However, a SPOF may occur on management nodes running in centralized mode, which becomes the weakness of the overall system reliability.

  MRS provides similar double-node mechanisms for all management nodes of the service components, such as Manager, Presto, HDFS NameNodes, Hive Servers, HBase HMasters, YARN Resource Managers, Kerberos Servers, and Ldap Servers. All of them are deployed in active/standby mode or configured with load sharing, effectively preventing SPOFs from affecting system reliability.

- Comprehensive reliability mechanism

  By reliability analysis, the following measures to handle software and hardware exceptions are provided to improve the system reliability:

  - After power supply is restored, services are running properly regardless of a power failure of a single node or the whole cluster, ensuring data reliability in case of unexpected power failures. Key data will not be lost unless the hard disk is damaged.
  - Health status checks and fault handling of the hard disk do not affect services.
  - The file system faults can be automatically handled, and affected services can be automatically restored.
  - The process and node faults can be automatically handled, and affected services can be automatically restored.
  - The network faults can be automatically handled, and affected services can be automatically restored.

8. **MRS provides a visualized big data cluster management interface in a unified manner, making O&M easier.**

   - On the big data cluster management interface, service startup and stopping, configuration modification, and health check are available. MRS also provides visualized and convenient cluster management, monitoring, and alarm functions. Additionally, you can check and audit the system health status in one click, ensuring normal system running and lowering system O&M costs.
   - After Simple Message Notification (SMN) is configured, MRS can send real-time cluster health status information, including cluster changes and component alarms in real time to you through SMS messages or emails, facilitating O&M, real-time monitoring, and real-time alarm sending.
   - MRS supports rolling patch upgrade and provides visualized patch release information and one-click patch installation without manual intervention, ensuring long-term stability of user clusters.
   - If a problem occurs when you use an MRS cluster, you can initiate O&M authorization on the MRS management console. O&M personnel can

help you quickly locate the problem, and you can revoke the authorization at any time. You can also initiate log sharing on the MRS management console to share a specified log scope with O&M personnel, so that O&M personnel can locate faults without accessing the cluster.

- MRS supports to dump logs about cluster creation failures to OBS for O&M personnel to obtain and analyze the logs.

9. **MRS has an open ecosystem and supports seamless interconnection with peripheral services, allowing you to quickly build a unified big data platform.**
   - Based on MRS, a full-stack big data service, enterprises can build a unified big data platform for data access, data storage, data analysis, and value mining, and interconnect with Data Lake Governance Center (DGC) and data visualization services to help customers easily resolve difficulties in data channel cloudification, big data job development and scheduling, and data display. Thereby, customers are free from complex big data platform construction and professional big data optimization and maintenance, and they can focus on industry applications and use one piece of data in multiple service scenarios. DGC is a one-stop operation platform for entire data lifecycle management. It provides data integration, data development, data governance, data service, and data visualization functions. MRS data can be connected to DGC, which greatly reduces the threshold for using big data. Based on the visualized development GUI, diverse data development types (script development and job development), fully hosted job scheduling and O&M monitoring capabilities, and built-in industry data processing pipelines, it helps you quickly build big data processing centers with visualized development processes and online collaborative operations. In this way, data can be well-managed and scheduled for real profits quicker.
   - MRS is fully compatible with the open source big data ecosystem. With abundant data and application migration tools, MRS helps you quickly migrate data from your own platforms without code modification and service interruption.

1.3 Application Scenarios

Big data is ubiquitous in people's lives. MRS is suitable to process big data in the industries such as the Internet of things (IoT), e-commerce, finance, manufacturing, healthcare, energy, and government departments.

Large-scale data analysis

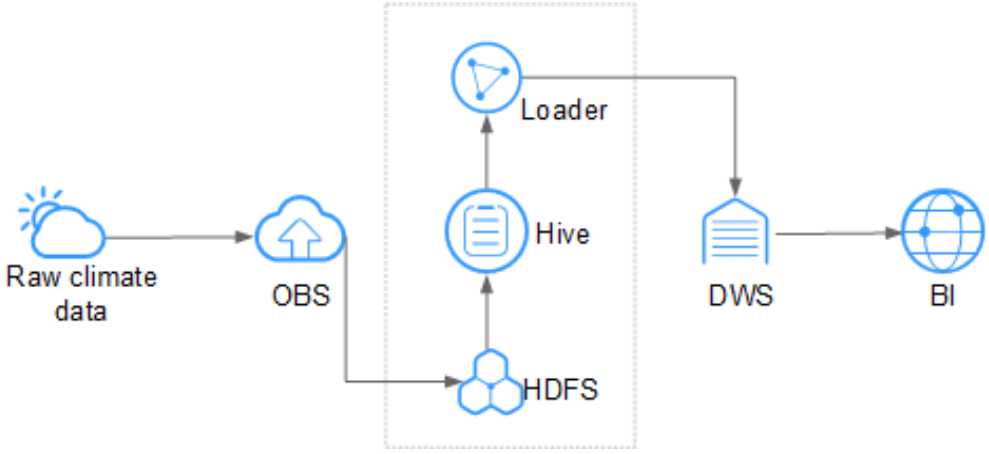
Large-scale data analysis is a major scenario in modern big data systems. Generally, an enterprise has multiple data sources. After data is accessed,extract, transform, and load (ETL) processing is required to generate modelized data for each service module to analyze and sort out data. This type of service has the following characteristics:

- The requirements for real-time execution are not high, and job execution time ranges from dozens of minutes to hours.
- The data volume is large.

- There are various data sources and diversified formats.
- Data processing usually consists of multiple tasks, and resources need to be planned in detail.

In the environmental protection industry, climate data is stored on OBS and periodically dumped into HDFS for batch analysis. 10 TB of climate data can be analyzed in 1 hour.

Figure 1-2 Large-scale data analysis in the environmental protection industry



MRS has the following advantages in this scenario.

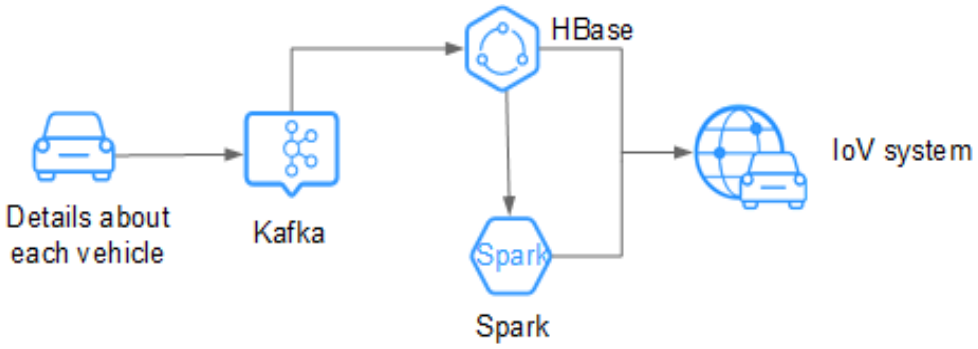
- Low cost: OBS offers cost-effective storage.
- Massive data analysis: TB/PB-level data is analyzed by Hive.
- Visualized data import and export tool: Loader exports data to Data Warehouse Service (DWS) for business intelligence (BI) analysis.

Large-scale data storage

A user who has a large amount of structured data usually requires index-based quasi-real-time query capabilities. For example, in an Internet of Vehicles (IoV) scenario, vehicle maintenance information is queried by vehicle number. Therefore, vehicle information is indexed based on vehicle numbers when it is being stored, to implement second-level response in this scenario. Generally, the data volume is large. The user may store data for one to three years.

For example, in the IoV industry, an automobile company stores data on HBase, which supports PB-level storage and CDR queries in milliseconds.

Figure 1-3 Large-scale data storage in the IoV industry



MRS has the following advantages in this scenario.

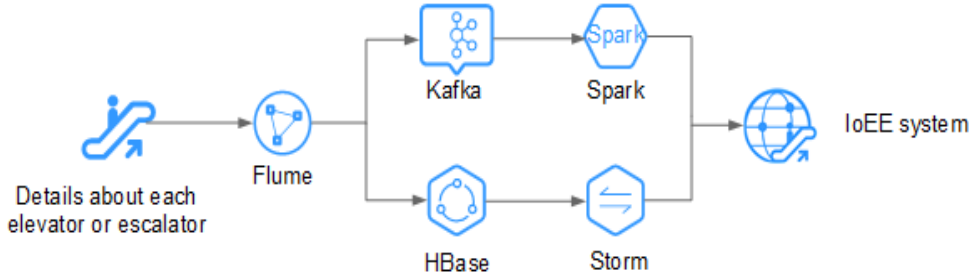
- Real time: Kafka accesses massive amounts of vehicle messages in real time.
- Massive data storage: HBase stores massive volumes of data and supports data queries in milliseconds.
- Distributed data query: Spark analyzes and queries massive volumes of data.

Real-time data processing

Real-time data processing is usually used in scenarios such as anomaly detection, fraud detection, rule-based alarming, and service process monitoring. Data is processed while it is being inputted to the system.

For example, in the Internet of elevators & escalators (IoEE) industry, data of smart elevators and escalators is imported to MRS streaming clusters in real time for real-time alarming.

Figure 1-4 Low-latency streaming processing in the IoEE industry



MRS has the following advantages in this scenario.

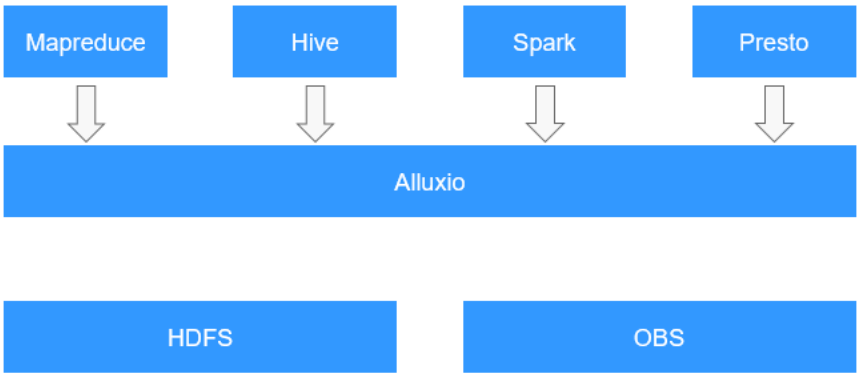
- Real-time data ingestion: Flume implements real-time data ingestion and provides various data collection and storage access methods.
- Data source access: Kafka accesses data of tens of thousands of elevators and escalators in real time.

1.4 Components

1.4.1 Alluxio

Alluxio is data orchestration technology for analytics and AI for the cloud. In the MRS big data ecosystem, Alluxio lies between computing and storage. It provides a data abstraction layer for computing frameworks including Apache Spark, Presto, MapReduce, and Apache Hive, so that upper-layer computing applications can access persistent storage systems including HDFS and OBS through unified client APIs and a global namespace. In this way, computing and storage are separated.

Figure 1-5 Alluxio architecture



Advantages:

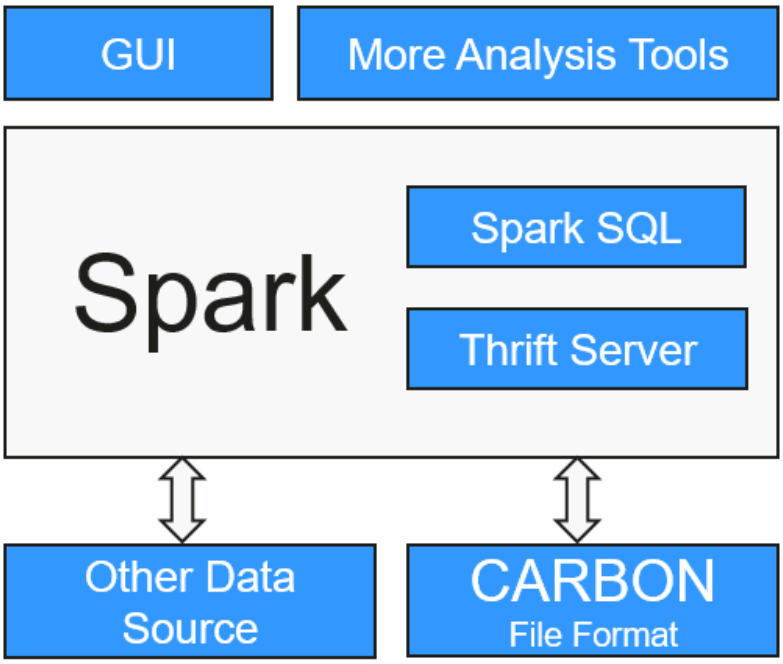
- Provides in-memory I/O throughput, and makes elastically scale data-driven applications cost effective.
- Simplified cloud and object storage access
- Simplified data management and a single point of access to multiple data sources
- Easy application deployment

For details about Alluxio, visit **https://docs.alluxio.io/os/user/stable/en/Overview.html**.

1.4.2 CarbonData

CarbonData is a new Apache Hadoop native data-store format. CarbonData allows faster interactive queries over PetaBytes of data using advanced columnar storage, index, compression, and encoding techniques to improve computing efficiency. In addition, CarbonData is also a high-performance analysis engine that integrates data sources with Spark.

Figure 1-6 Basic architecture of CarbonData



The purpose of using CarbonData is to provide quick response to ad hoc queries of big data. Essentially, CarbonData is an Online Analytical Processing (OLAP) engine, which stores data using tables similar to those in Relational Database Management System (RDBMS). You can import more than 10 TB data to tables created in CarbonData format, and CarbonData automatically organizes and stores data using the compressed multi-dimensional indexes. After data is loaded to CarbonData, CarbonData responds to ad hoc queries in seconds.

CarbonData integrates data sources into the Spark ecosystem. You can use Spark SQL to query and analyze data, or use the third-party tool ThriftServer provided by Spark to connect to Spark SQL.

CarbonData features

- SQL: CarbonData is compatible with Spark SQL and supports SQL query operations performed on Spark SQL.
- Simple Table dataset definition: CarbonData allows you to define and create datasets by using user-friendly Data Definition Language (DDL) statements. CarbonData DDL is flexible and easy to use, and can define complex tables.
- Easy data management: CarbonData provides various data management functions for data loading and maintenance. It can load historical data and incrementally load new data. The loaded data can be deleted according to the loading time and specific data loading operations can be canceled.
- CarbonData file format is a columnar store in HDFS. It has many features that a modern columnar format has, such as splittable and compression schema.

Unique features of CarbonData

- Stores data along with index: Significantly accelerates query performance and reduces the I/O scans and CPU resources, when there are filters in the query. CarbonData index consists of multiple levels of indices. A processing

framework can leverage this index to reduce the task it needs to schedule and process, and it can also perform skip scan in more finer grain unit (called blocklet) in task side scanning instead of scanning the whole file.

- Operable encoded data: Through supporting efficient compression and global encoding schemes, CarbonData can query on compressed/encoded data. The data can be converted just before returning the results to the users, which is "late materialized".
- Supports various use cases with one single data format: like interactive OLAP-style query, Sequential Access (big scan), and Random Access (narrow scan).

Key technologies and advantages of CarbonData

- Quick query response: CarbonData features high-performance query. The query speed of CarbonData is 10 times of that of Spark SQL. It uses dedicated data formats and applies multiple index technologies, global dictionary code, and multiple push-down optimizations, providing quick response to TB-level data queries.
- Efficient data compression: CarbonData compresses data by combining the lightweight and heavyweight compression algorithms. This significantly saves 60% to 80% data storage space and the hardware storage cost.

For details about CarbonData architecture and principles, see **https://carbondata.apache.org/**.

1.4.3 ClickHouse

Introduction to ClickHouse

ClickHouse is an open-source columnar database oriented to online analysis and processing. It is independent of the Hadoop big data system and features ultimate compression rate and fast query performance. In addition, ClickHouse supports SQL query and provides good query performance, especially the aggregation analysis and query performance based on large and wide tables. The query speed is one order of magnitude faster than that of other analytical databases.

The core functions of ClickHouse are as follows:

Comprehensive DBMS functions

ClickHouse has comprehensive database management functions, including the basic functions of a Database Management System (DBMS):

- Data Definition Language (DDL): allows databases, tables, and views to be dynamically created, modified, or deleted without restarting services.
- Data Manipulation Language (DML): allows data to be queried, inserted, modified, or deleted dynamically.
- Permission control: supports user-based database or table operation permission settings to ensure data security.
- Data backup and restoration: supports data backup, export, import, and restoration to meet the requirements of the production environment.
- Distributed management: provides the cluster mode to automatically manage multiple database nodes.

Column-based storage and data compression

ClickHouse is a database that uses column-based storage. Data is organized by column. Data in the same column is stored together, and data in different columns is stored in different files.

During data query, columnar storage can reduce the data scanning range and data transmission size, thereby improving data query efficiency.

In a traditional row-based database system, data is stored in the sequence in **Table 1-1**:

Table 1-1 Row-based database

row	ID	Flag	Name	Event	Time
0	12345678901	0	name1	1	2020/1/11 15:19
1	32345678901	1	name2	1	2020/5/12 18:10
2	42345678901	1	name3	1	2020/6/13 17:38
N	...	...	...	...	...

In a row-based database, data in the same row is physically stored together. In a column-based database system, data is stored in the sequence in **Table 1-2**:

Table 1-2 Columnar database

row:	0	1	2	N
ID:	12345678901	32345678901	42345678901	...
Flag:	0	1	1	...
Name:	name1	name2	name3	...
Event:	1	1	1	...
Time:	2020/1/11 15:19	2020/5/12 18:10	2020/6/13 17:38	...

This example shows only the arrangement of data in a columnar database. Columnar databases store data in the same column together and data in different columns separately. Columnar databases are more suitable for online analytical processing (OLAP) scenarios.

Vectorized executor

ClickHouse uses CPU's Single Instruction Multiple Data (SIMD) to implement vectorized execution. SIMD is an implementation mode that uses a single instruction to operate multiple pieces of data and improves performance with data

parallelism (other methods include instruction-level parallelism and thread-level parallelism). The principle of SIMD is to implement parallel data operations at the CPU register level.

Relational model and SQL query

ClickHouse uses SQL as the query language and provides standard SQL query APIs for existing third-party analysis visualization systems to easily integrate with ClickHouse.

In addition, ClickHouse uses a relational model. Therefore, the cost of migrating the system built on a traditional relational database or data warehouse to ClickHouse is lower.

Data sharding and distributed query

The ClickHouse cluster consists of one or more shards, and each shard corresponds to one ClickHouse service node. The maximum number of shards depends on the number of nodes (one shard corresponds to only one service node).

ClickHouse introduces the concepts of local table and distributed table. A local table is equivalent to a data shard. A distributed table itself does not store any data. It is an access proxy of the local table and functions as the sharding middleware. With the help of distributed tables, multiple data shards can be accessed by using the proxy, thereby implementing distributed query.

ClickHouse Applications

ClickHouse is short for Click Stream and Data Warehouse. It is initially applied to a web traffic analysis tool to perform OLAP analysis for data warehouses based on page click event flows. Currently, ClickHouse is widely used in Internet advertising, app and web traffic analysis, telecommunications, finance, and Internet of Things (IoT) fields. It is applicable to business intelligence application scenarios and has a large number of applications and practices worldwide. For details, visit **https://clickhouse.tech/docs/en/introduction/adopters/**.

ClickHouse Enhanced Open Source Features

MRS ClickHouse has advantages such as automatic cluster mode, HA deployment, and smooth and elastic scaling.

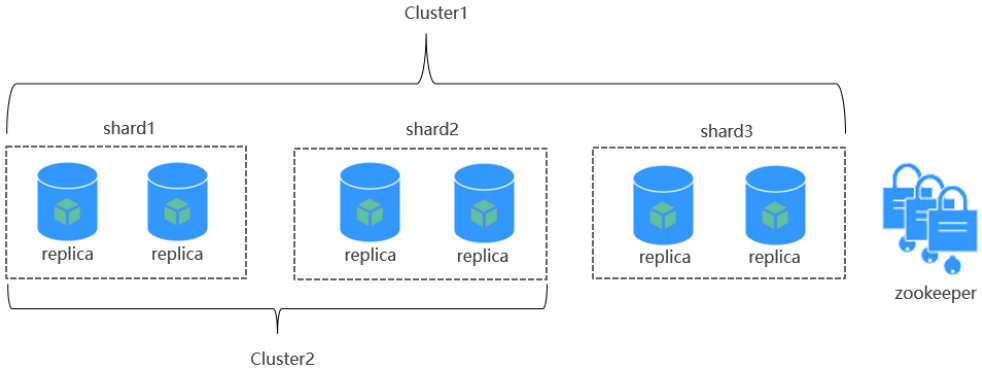
- Automatic Cluster Mode

  As shown in **Figure 1-7**, a cluster consists of multiple ClickHouse nodes, which has no central node. It is more of a static resource pool. If the ClickHouse cluster mode is used for services, you need to pre-define the cluster information in the configuration file of each node. Only in this way, services can be correctly accessed.

  Figure 1-7 ClickHouse cluster

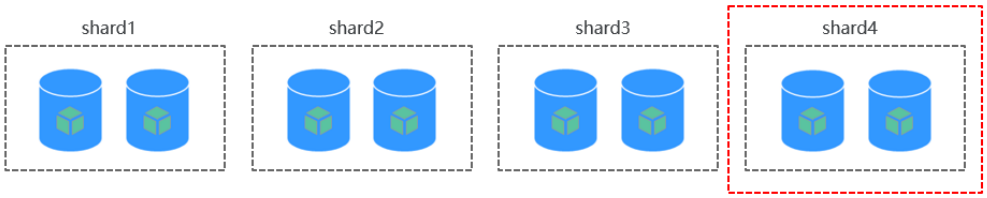
Users are unaware of data partitions and replica storage in common database systems. However, ClickHouse allows you to proactively plan and define detailed configurations such as shards, partitions, and replica locations. The ClickHouse instance of MRS packs the work in a unified manner and adapts it to the automatic mode, implementing unified management, which is flexible and easy to use. A ClickHouse instance consists of three ZooKeeper nodes and multiple ClickHouse nodes. The Dedicated Replica mode is used to ensure high reliability of dual data copies.

Figure 1-8 ClickHouse cluster structure



- Smooth and Elastic Scaling

  With the rapid growth of services, MRS provides smooth and elastic scaling capabilities to quickly meet service growth requirements in scenarios where the cluster storage capacity or CPU computing resources are not enough. When you expand the capacity of ClickHouse nodes in a cluster, MRS provides a one-click data balancing tool and gives you the initiative to balance data. You can determine the data balancing mode and time based on service characteristics to ensure service availability, implementing smooth scaling.

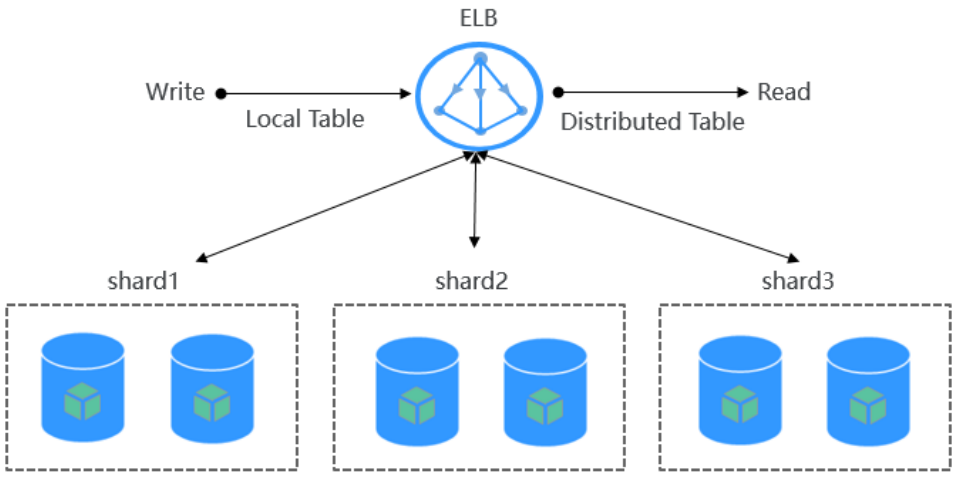


- HA Deployment Architecture

  MRS uses the ELB-based high availability (HA) deployment architecture to automatically distribute user access traffic to multiple backend nodes, expanding service capabilities to external systems and improving fault tolerance. As shown in **Figure 1-9**, when a client application requests a cluster, Elastic Load Balance (ELB) is used to distribute traffic. With the ELB polling mechanism, data is written to local tables and read from distributed tables on different nodes. In this way, data read/write load and high availability of application access are guaranteed.

  After the ClickHouse cluster is provisioned, each ClickHouse instance node in the cluster corresponds to a replica, and two replicas form a logical shard. For example, when creating a ReplicatedMergeTree table, you can specify shards so that data can be automatically synchronized between two replicas in the same shard.

Figure 1-9 HA deployment architecture



1.4.4 DBService

1.4.4.1 DBService Basic Principles

Overview

DBService is a HA storage system for relational databases, which is applicable to the scenario where a small amount of data (about 10 GB) needs to be stored, for example, component metadata. DBService can only be used by internal components of a cluster and provides data storage, query, and deletion functions.

DBService is a basic component of a cluster. Components such as Hive, Hue, Oozie, Loader, and Redis, and Loader store their metadata in DBService, and provide the metadata backup and restoration functions by using DBService.

DBService Architecture

DBService in the cluster works in active/standby mode. Two DBServer instances are deployed and each instance contains three modules: HA, Database, and FloatIP.

Figure 1-10 shows the DBService logical architecture.

Figure 1-10 DBService architecture

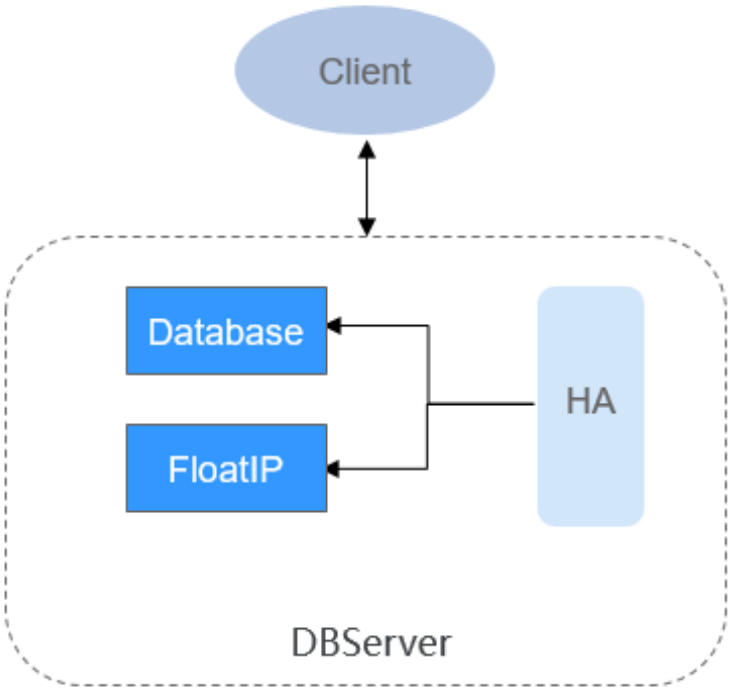


Table 1-3 describes the modules shown in **Figure 1-10**

Table 1-3 Module description

Name	Description
HA	HA management module. The active/standby DBServer uses the HA module for management.
Database	Database module. This module stores the metadata of the Client module.
FloatIP	Floating IP address that provides the access function externally. It is enabled only on the active DBServer instance and is used by the Client module to access Database.
Client	Client using the DBService component, which is deployed on the component instance node. The client connects to the database by using FloatIP and then performs metadata adding, deleting, and modifying operations.

1.4.4.2 Relationship Between DBService and Other Components

DBService is a basic component of a cluster. Components such as Hive, Hue, Oozie, Loader, Metadata, and Redis, and Loader store their metadata in DBService, and provide the metadata backup and restoration functions by using DBService.

1.4.5 Flink

1.4.5.1 Flink Basic Principles

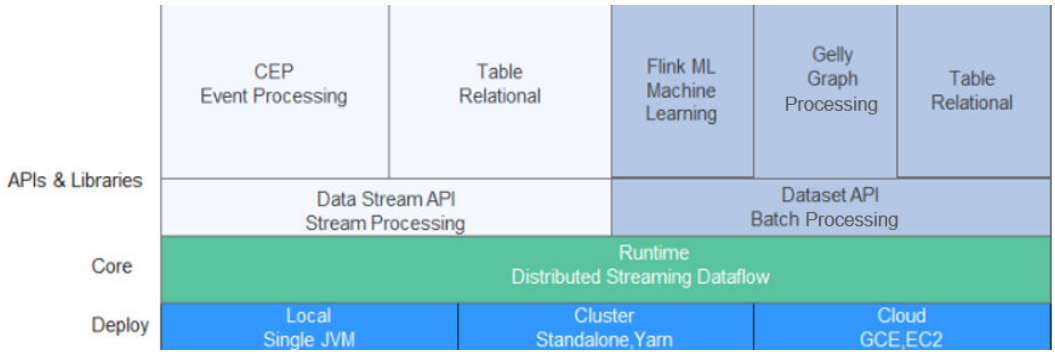
Overview

Flink is a unified computing framework that supports both batch processing and stream processing. It provides a stream data processing engine that supports data distribution and parallel computing. Flink features stream processing and is a top open source stream processing engine in the industry.

Flink provides high-concurrency pipeline data processing, millisecond-level latency, and high reliability, making it extremely suitable for low-latency data processing.

Figure 1-11 shows the technology stack of Flink.

Figure 1-11 Technology stack of Flink



Flink provides the following features in the current version:

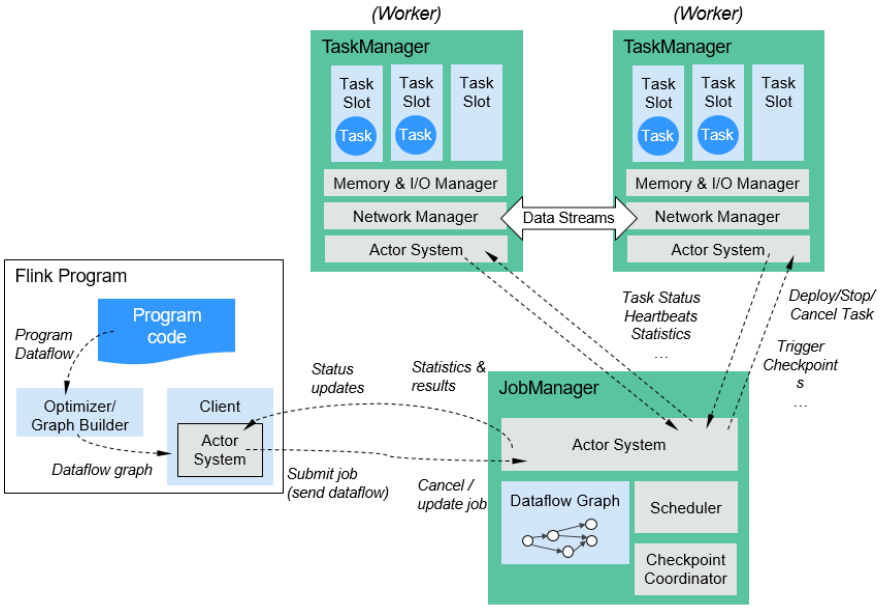
- DataStream
- Checkpoint
- Window
- Job Pipeline
- Configuration Table

Other features are inherited from the open source community and are not enhanced. For details, visit **https://ci.apache.org/projects/flink/flink-docs-release-1.12/**.

Flink Architecture

Figure 1-12 shows the Flink architecture.

Figure 1-12 Flink architecture



As shown in the above figure, the entire Flink system consists of three parts:

- Client

  Flink client is used to submit jobs (streaming jobs) to Flink.

- TaskManager

  TaskManager is a service execution node of Flink. It executes specific tasks. A Flink system can have multiple TaskManagers. These TaskManagers are equivalent to each other.

- JobManager

  JobManager is a management node of Flink. It manages all TaskManagers and schedules tasks submitted by users to specific TaskManagers. In high-availability (HA) mode, multiple JobManagers are deployed. Among these JobManagers, one is selected as the active JobManager, and the others are standby.

For more information about the Flink architecture, visit **https://ci.apache.org/projects/flink/flink-docs-master/docs/concepts/flink-architecture/**.

Flink Principles

- **Stream & Transformation & Operator**

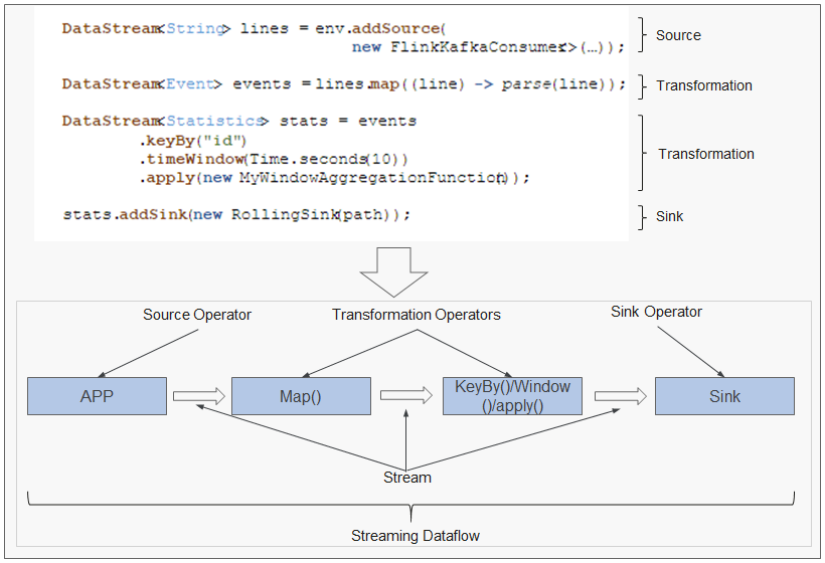
  A Flink program consists of two building blocks: stream and transformation.

  a. Conceptually, a stream is a (potentially never-ending) flow of data records, and a transformation is an operation that takes one or more streams as input, and produces one or more output streams as a result.

  b. When a Flink program is executed, it is mapped to a streaming dataflow. A streaming dataflow consists of a group of streams and transformation operators. Each dataflow starts with one or more source operators and ends in one or more sink operators. A dataflow resembles a directed acyclic graph (DAG).

Figure 1-13 shows the streaming dataflow to which a Flink program is mapped.

Figure 1-13 Example of Flink DataStream



As shown in Figure 1-13, **FlinkKafkaConsumer** is a source operator; Map, KeyBy, TimeWindow, and Apply are transformation operators; RollingSink is a sink operator.

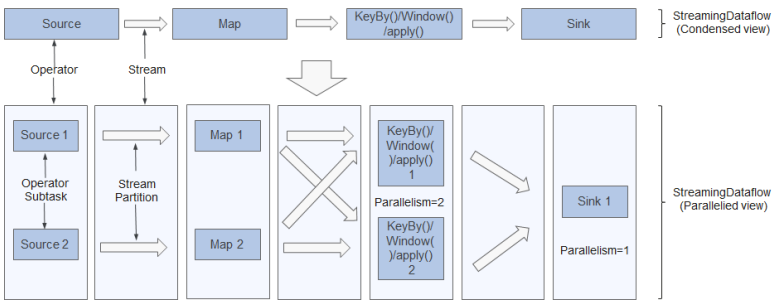
- **Pipeline Dataflow**

  Applications in Flink can be executed in parallel or distributed modes. A stream can be divided into one or more stream partitions, and an operator can be divided into multiple operator subtasks.

  The executor of streams and operators are automatically optimized based on the density of upstream and downstream operators.

  – Operators with low density cannot be optimized. Each operator subtask is separately executed in different threads. The number of operator subtasks is the parallelism of that particular operator. The parallelism (the total number of partitions) of a stream is that of its producing operator. Different operators of the same program may have different levels of parallelism, as shown in Figure 1-14.

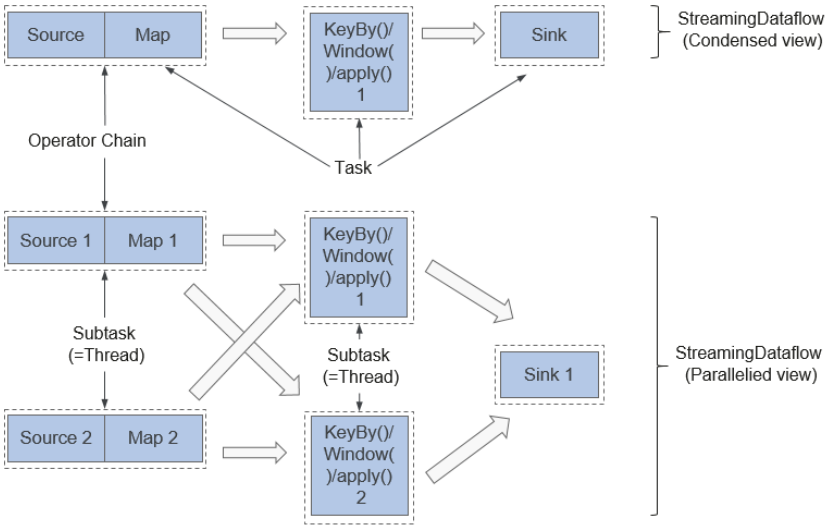
  Figure 1-14 Operator

  – Operators with high density can be optimized. Flink chains operator subtasks together into a task, that is, an operator chain. Each operator

chain is executed by one thread on TaskManager, as shown in **Figure 1-15**.

Figure 1-15 Operator chain



- In the upper part of **Figure 1-15**, the condensed Source and Map operators are chained into an Operator Chain, that is, a larger operator. The Operator Chain, KeyBy, and Sink all represent an operator respectively and are connected with each other through streams. Each operator corresponds to one task during the running. Namely, there are three tasks in the upper part.
- In the lower part of **Figure 1-15**, each task, except Sink, is paralleled into two subtasks. The parallelism of the Sink operator is one.

Key Features

- Stream processing

  The real-time stream processing engine features high throughput, high performance, and low latency, which can provide processing capability within milliseconds.

- Various status management

  The stream processing application needs to store the received events or intermediate result in a certain period of time for subsequent access and processing at a certain time point. Flink provides diverse features for status management, including:

  - Multiple basic status types: Flink provides various states for data structures, such as ValueState, ListState, and MapState. Users can select the most efficient and suitable status type based on the service model.
  - Rich State Backend: State Backend manages the status of applications and performs Checkpoint operations as required. Flink provides different State Backends. State can be stored in the memory or RocksDB, and supports the asynchronous and incremental Checkpoint mechanism.

  - Exactly-once state consistency: The Checkpoint and fault recovery capabilities of Flink ensure that the application status of tasks is consistent before and after a fault occurs. Flink supports transactional output for some specific storage devices. In this way, exactly-once output can be ensured even when a fault occurs.

- Various time semantics

  Time is an important part of stream processing applications. For real-time stream processing applications, operations such as window aggregation, detection, and matching based on time semantics are very common. Flink provides various time semantics.

  - Event-time: The timestamp provided by the event is used for calculation, making it easier to process the events that arrive at a random sequence or arrive late.
  - Watermark: Flink introduces the concept of Watermark to measure the development of event time. Watermark also provides flexible assurance for balancing processing latency and data integrity. When processing event streams with Watermark, Flink provides multiple processing options if data arrives after the calculation, for example, redirecting data (side output) or updating the calculation result.
  - Processing-time and Ingestion-time are supported.
  - Highly flexible streaming window: Flink supports the time window, count window, session window, and data-driven customized window. You can customize the triggering conditions to implement the complex streaming calculation mode.

- Fault tolerance mechanism

  In a distributed system, if a single task or node breaks down or is faulty, the entire task may fail. Flink provides a task-level fault tolerance mechanism, which ensures that user data is not lost when an exception occurs in a task and can be automatically restored.

  - Checkpoint: Flink implements fault tolerance based on checkpoint. Users can customize the checkpoint policy for the entire task. When a task fails, the task can be restored to the status of the latest checkpoint and data after the snapshot is resent from the data source.
  - Savepoint: A savepoint is a consistent snapshot of application status. The savepoint mechanism is similar to that of checkpoint. However, the savepoint mechanism needs to be manually triggered. The savepoint mechanism ensures that the status information of the current stream application is not lost during task upgrade or migration, facilitating task suspension and recovery at any time point.

- Flink SQL

  Table APIs and SQL use Apache Calcite to parse, verify, and optimize queries. Table APIs and SQL can be seamlessly integrated with DataStream and DataSet APIs, and support user-defined scalar functions, aggregation functions, and table value functions. The definition of applications such as data analysis and ETL is simplified. The following code example shows how to use Flink SQL statements to define a counting application that records session times.

  ```
  SELECT userId, COUNT(*)
  FROM clicks
  GROUP BY SESSION(clicktime, INTERVAL '30' MINUTE), userId
  ```

For more information about Flink SQL, see **https://ci.apache.org/projects/flink/flink-docs-master/dev/table/sqlClient.html**.

- CEP in SQL

  Flink allows users to represent complex event processing (CEP) query results in SQL for pattern matching and evaluate event streams on Flink.

  CEP SQL is implemented through the **MATCH_RECOGNIZE** SQL syntax. The **MATCH_RECOGNIZE** clause is supported by Oracle SQL since Oracle Database 12c and is used to indicate event pattern matching in SQL. The following is an example of CEP SQL:

```
SELECT T.aid, T.bid, T.cid
FROM MyTable
   MATCH_RECOGNIZE (
     PARTITION BY userid
     ORDER BY proctime
     MEASURES
       A.id AS aid,
       B.id AS bid,
       C.id AS cid
     PATTERN (A B C)
     DEFINE
       A AS name = 'a',
       B AS name = 'b',
       C AS name = 'c'
   ) AS T
```

1.4.5.2 Flink HA Solution

Flink HA Solution

A Flink cluster has only one JobManager. This has the risks of single point of failures (SPOFs). There are three modes of Flink: Flink On Yarn, Flink Standalone, and Flink Local. Flink On Yarn and Flink Standalone modes are based on clusters and Flink Local mode is based on a single node. Flink On Yarn and Flink Standalone provide an HA mechanism. With such a mechanism, you can recover the JobManager from failures and thereby eliminate SPOF risks. This section describes the HA mechanism of the Flink On Yarn.

Flink supports the HA mode and job exception recovery that highly depend on ZooKeeper. If you want to enable the two functions, configure ZooKeeper in the **flink-conf.yaml** file in advance as follows:

```
high-availability: zookeeper
high-availability.zookeeper.quorum:  ZooKeeper IP address:2181
high-availability.storageDir: hdfs:///flink/recovery
```

Flink On Yarn

Flink JobManager and Yarn ApplicationMaster are in the same process. Yarn ResourceManager monitors ApplicationMaster. If ApplicationMaster is abnormal, Yarn restarts it and restores all JobManager metadata from HDFS. During the recovery, existing tasks cannot run and new tasks cannot be submitted. ZooKeeper stores JobManager metadata, such as information about jobs, to be used by the new JobManager. A TaskManager failure is listened and processed by the DeathWatch mechanism of Akka on JobManager. When a TaskManager fails, a container is requested again from Yarn and a TaskManager is created.

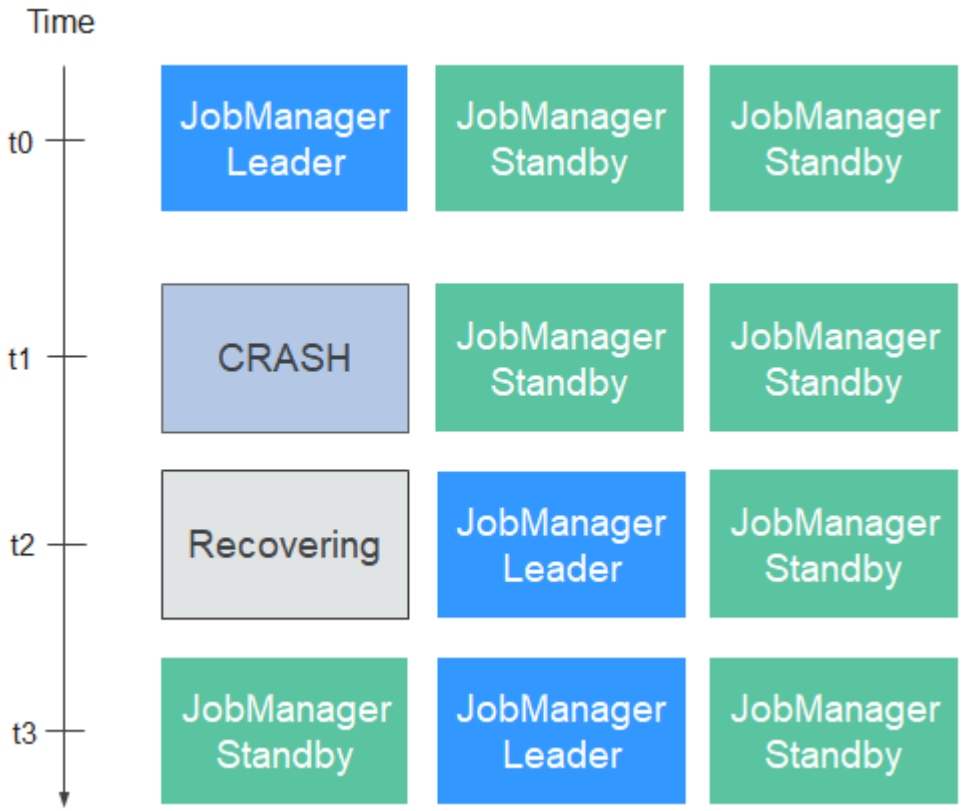
For more information about the HA solution of Flink on Yarn, visit **https://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-site/ResourceManagerHA.html**.

For details about how to set **yarn-site.xml**, visit **https://ci.apache.org/projects/flink/flink-docs-release-1.12/ops/jobmanager_high_availability.html**.

Standalone

In the standalone mode, multiple JobManagers can be started and ZooKeeper elects one as the leader JobManager. In this mode, there is a leader JobManager and multiple standby JobManagers. If the leader JobManager fails, a standby JobManager takes over the leadership. **Figure 1-16** shows the process of a leader/standby JobManager switchover.

Figure 1-16 Switchover process



Restoring TaskManager

A TaskManager failure is listened and processed by the DeathWatch mechanism of Akka on JobManager. If the TaskManager fails, the JobManager creates a TaskManager and migrates services to the created TaskManager.

Restoring JobManager

Flink JobManager and Yarn ApplicationMaster are in the same process. Yarn ResourceManager monitors ApplicationMaster. If ApplicationMaster is abnormal, Yarn restarts it and restores all JobManager metadata from HDFS. During the recovery, existing tasks cannot run and new tasks cannot be submitted.

Restoring Jobs

If you want to restore jobs, ensure that the startup policy is configured in Flink configuration files. Supported restart policies are **fixed-delay**, **failure-rate**, and **none**. Jobs can be restored only when the policy is configured to **fixed-delay** or **failure-rate**. If the restart policy is configured to **none** and checkpoint is configured for jobs, the restart policy is automatically configured to **fixed-delay** and the value of **restart-strategy.fixed-delay.attempts** (which specifies the number of retry times) is configured to **Integer.MAX_VALUE**.

For details about the three strategies, visit **https://ci.apache.org/projects/flink/flink-docs-release-1.12/dev/task_failure_recovery.html**. The following is an example of the restart policy configuration:

```
restart-strategy: fixed-delay
restart-strategy.fixed-delay.attempts: 3
restart-strategy.fixed-delay.delay: 10 s
```

Jobs will be restored in the following scenarios:

- If a JobManager fails, all its jobs are stopped, and will be recovered after another JobManager is created and running.
- If a TaskManager fails, all tasks on the TaskManager are stopped, and will be started until there are available resources.
- When a task of a job fails, the job is restarted.

  NOTE

  For details about how to configure the restart policy of a job, visit **https://ci.apache.org/projects/flink/flink-docs-release-1.12/ops/jobmanager_high_availability.html**.

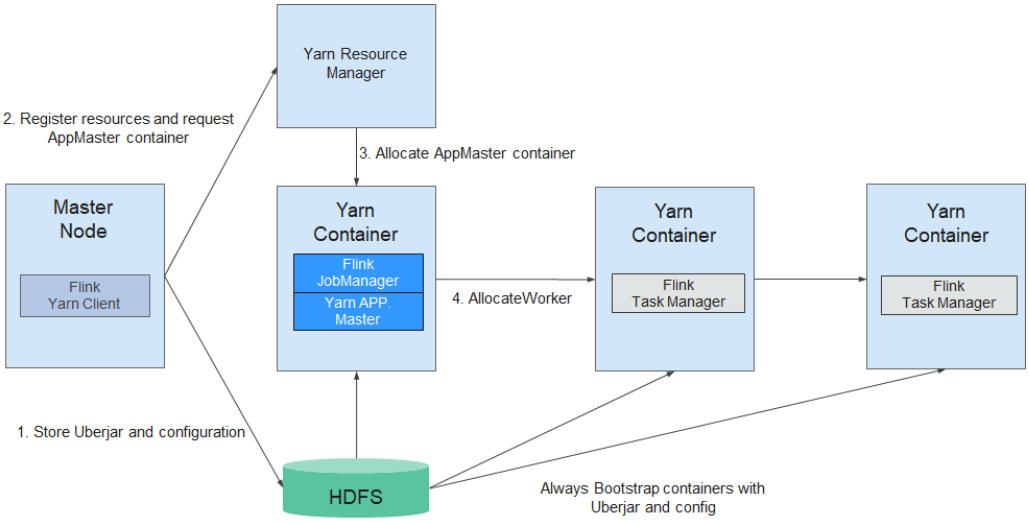
1.4.5.3 Relationship with Other Components

Relationship between Flink and Yarn

Flink supports Yarn-based cluster management mode. In this mode, Flink serves as an application of Yarn and runs on Yarn.

Figure 1-17 shows how Flink interacts with Yarn.

Figure 1-17 Flink interaction with Yarn



1. The Flink Yarn Client first checks whether there are sufficient resources for starting the Yarn cluster. If yes, the Flink Yarn client uploads JAR packages and configuration files to HDFS.
2. Flink Yarn client communicates with Yarn ResourceManager to request a container for starting ApplicationMaster. After all Yarn NodeManagers finish downloading the JAR package and configuration files, the ApplicationMaster is started.
3. During the startup, the ApplicationMaster interacts with the Yarn ResourceManager to request the container for starting a TaskManager. After the container is ready, the TaskManager process is started.
4. In the Flink Yarn cluster, the ApplicationMaster and Flink JobManager are running in the same container. The ApplicationMaster informs each TaskManager of the RPC address of the JobManager. After TaskManagers are started successfully, they register with the JobManager.
5. After all TaskManagers have registered with the JobManager successfully, Flink starts up in the Yarn cluster. Then, the Flink Yarn client can submit Flink jobs to the JobManager, and Flink can perform mapping, scheduling, and computing for the jobs.

1.4.5.4 Flink Enhanced Open Source Features

1.4.5.4.1 Window

Enhanced Open Source Feature: Window

This section describes the sliding window of Flink and provides the sliding window optimization method. For details about windows, visit **https://ci.apache.org/projects/flink/flink-docs-release-1.12/dev/stream/operators/windows.html**.

Introduction to Window

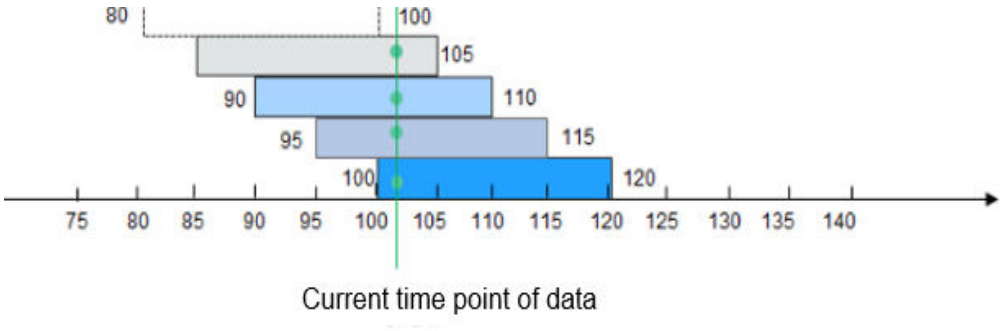
Data in a window is saved as intermediate results or original data. If you perform a sum operation (**window(SlidingEventTimeWindows.of(Time.seconds(20), Time.seconds(5))).sum**) on data in the window, only the intermediate result will be retained. If a custom window (**window(SlidingEventTimeWindows.of(Time.seconds(20), Time.seconds(5))).apply(new UDF)**) is used, all original data in the window will be saved.

If custom windows **SlidingEventTimeWindow** and **SlidingProcessingTimeWindow** are used, data is saved as multiple backups. Assume that the window is defined as follows:

```
window(SlidingEventTimeWindows.of(Time.seconds(20), Time.seconds(5))).apply(new
UDFWindowFunction)
```

If a block of data arrives, it is assigned to four different windows (20/5 = 4). That is, the data is saved as four copies in the memory. When the window size or sliding period is set to a large value, data will be saved as excessive copies, causing redundancy.

Figure 1-18 Original structure of a window



If a data block arrives at the 102nd second, it is assigned to windows [85, 105), [90, 110), [95, 115), and [100, 120).

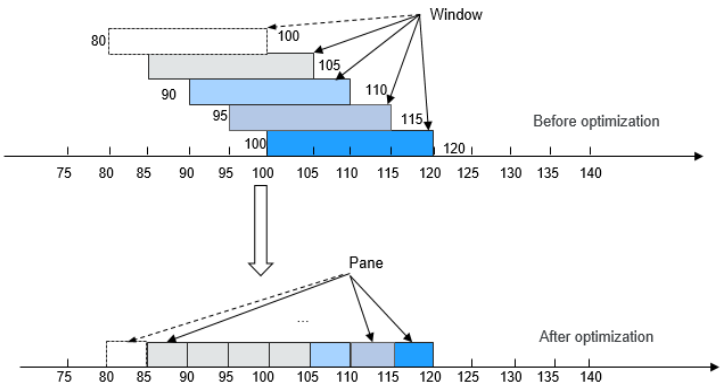
Window Optimization

As mentioned in the preceding, there are excessive data copies when original data is saved in SlidingEventTimeWindow and SlidingProcessingTimeWindow. To resolve this problem, the window that stores the original data is restructured, which optimizes the storage and greatly lowers the storage space. The window optimization scheme is as follows:

1. Use the sliding period as a unit to divide a window into different panes.

   A window consists of one or multiple panes. A pane is essentially a sliding period. For example, the sliding period (namely, the pane) of **window(SlidingEventTimeWindows.of(Time.seconds(20), Time.seconds.of(5)))** lasts for 5 seconds. If this window ranges from [100, 120), this window can be divided into panes [100, 105), [105, 110), [110, 115), and [115, 120).

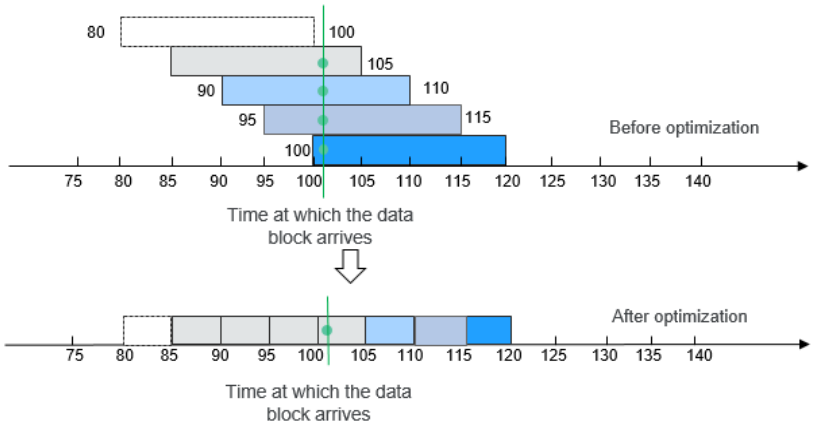
   Figure 1-19 Window optimization

2. When a data block arrives, it is not assigned to a specific window. Instead, Flink determines the pane to which the data block belongs based on the timestamp of the data block, and saves the data block into the pane.

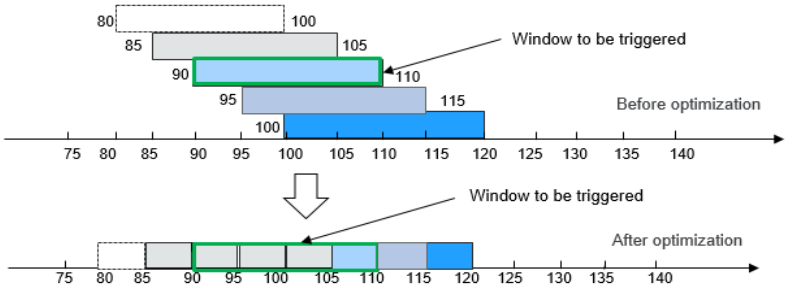
   A data block is saved only in one pane. In this case, only a data copy exists in the memory.

Figure 1-20 Saving data in a window



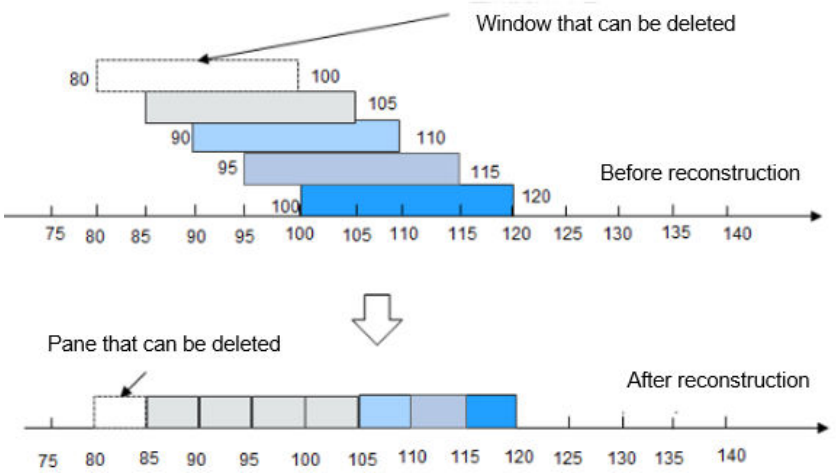
3. To trigger a window, compute all panes contained in the window, and combine all these panes into a complete window.

Figure 1-21 Triggering a window



4. If a pane is not required, you can delete it from the memory.

Figure 1-22 Deleting a window



After optimization, the quantity of data copies in the memory and snapshot is greatly reduced.

1.4.5.4.2 Job Pipeline

Enhanced Open Source Feature: Job Pipeline

Generally, logic code related to a service is stored in a large JAR package, which is called Fat JAR. Disadvantages of Fat JAR are as follows:

- When service logic becomes more and more complex, the size of the Fat JAR increases.
- Fat Jar makes coordination complex. Developers of all services are working with the same service logic. Even though the service logic can be divided into several modules, all modules are tightly coupled with each other. If the requirement needs to be changed, the entire flow diagram needs to be replanned.

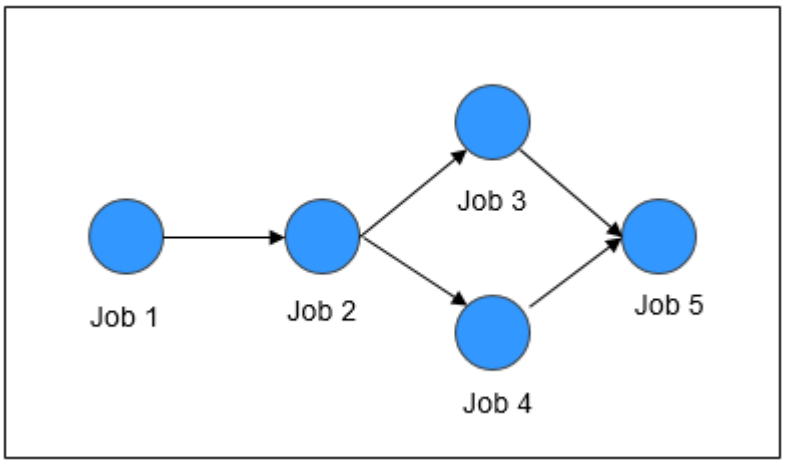
Splitting of jobs is facing the following problems:

- Data transmission between jobs can be achieved using Kafka. For example, job A transmits data to the topic A in Kafka, and then job B and job C read data from the topic A in Kafka. This solution is simple and easy to implement, but the latency is always longer than 100 ms.
- Operators are connected using the TCP protocol. In distributed environment, operators can be scheduled to any node and upstream and downstream services cannot detect the scheduling.

Job Pipeline

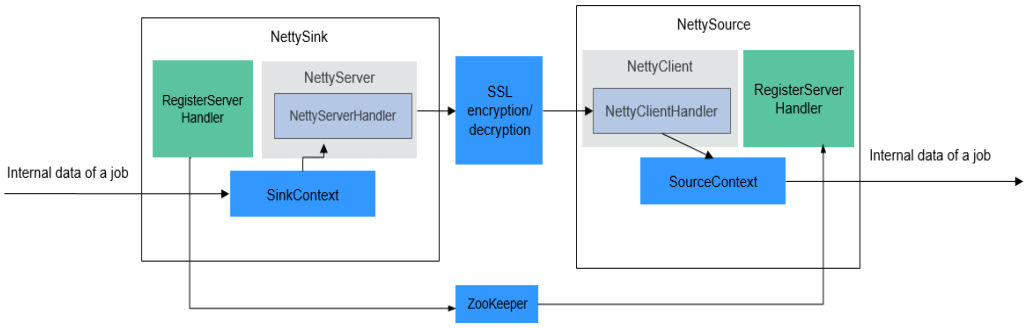
A pipeline consists of multiple Flink jobs connected through TCP. Upstream jobs can send data to downstream jobs. The flow diagram about data transmission is called a job pipeline, as shown in **Figure 1-23**.

Figure 1-23 Job pipeline



Job Pipeline Principles

Figure 1-24 Job pipeline principles



- NettySink and NettySource

  In a pipeline, upstream jobs and downstream jobs communicate with each other through Netty. The Sink operator of the upstream job works as a server and the Source operator of the downstream job works as a client. The Sink operator of the upstream job is called NettySink, and the Source operator of the downstream job is called NettySource.

- NettyServer and NettyClient

  NettySink functions as the server of Netty. In NettySink, NettyServer achieves the function of a server. NettySource functions as the client of Netty. In NettySource, NettyClient achieves the function of a client.

- Publisher

  The job that sends data to downstream jobs through NettySink is called a publisher.

- Subscriber

  The job that receives data from upstream jobs through NettySource is called a subscriber.

- RegisterServer

  RegisterServer is the third-party memory that stores the IP address, port number, and concurrency information about NettyServer.

- The general outside-in architecture is as follows:
  - NettySink->NettyServer->NettyServerHandler
  - NettySource->NettyClient->NettyClientHandler

Job Pipeline Functions

- **NettySink**

  NettySink consists of the following major modules:

  - RichParallelSinkFunction

    NettySink inherits RichParallelSinkFunction and attributes of Sink operators. The RichParallelSinkFunction API implements following functions:

    - Starts the NettySink operator.
    - Runs the NettySink operator and receives data from the upstream operator.

- Cancels the running of NettySink operators.

Following information can be obtained using the attribute of RichParallelSinkFunction:

- subtaskIndex about the concurrency of each NettySink operator.
- Concurrency of the NettySink operator.

– RegisterServerHandler

RegisterServerHandler interacts with the component of RegisterServer and defines following APIs:

- **start();**: Starts the RegisterServerHandler and establishes a contact with the third-party RegisterServer.
- **createTopicNode();**: Creates a topic node.
- **register();**: Registers information such as the IP address, port number, and concurrency to the topic node.
- **deleteTopicNode();**: Deletes a topic node.
- **unregister();**: Deletes registration information.
- **query();**: Queries registration information.
- **isExist();**: Verifies that a specific piece of information exists.
- **shutdown();**: Disables the RegisterServerHandler and disconnects from the third-party RegisterServer.

NOTE

- RegisterServerHandler API enables ZooKeeper to work as the handler of RegisterServer. You can customize your handler as required. Information is stored in ZooKeeper in the following form:

```
Namespace
|---Topic-1
  |---parallel-1
  |---parallel-2
  |....
  |---parallel-n
|---Topic-2
  |---parallel-1
  |---parallel-2
  |....
  |---parallel-m
|...
```

- Information about NameSpace can be obtained from the following parameters of the **flink-conf.yaml** file:

```
nettyconnector.registerserver.topic.storage: /flink/nettyconnector
```

- The simple authentication and security layer (SASL) authentication between ZookeeperRegisterServerHandler and ZooKeeper is implemented through the Flink framework.
- Ensure that each job has a unique topic. Otherwise, the subscription relationship may be unclear.
- When calling **shutdown()**, ZookeeperRegisterServerHandler deletes the registration information about the current concurrency, and then attempts to delete the topic node. If the topic node is not empty, deletion will be canceled, because not all concurrency has exited.

  - NettyServer

    NettyServer is the core of the NettySink operator, whose main function is to create a NettyServer and receive connection requests from NettyClient. Use NettyServerHandler to send data received from upstream operators of a same job. The port number and subnet of NettyServer needs to be configured in the **flink-conf.yaml** file.

    - Port range
      ```
      nettyconnector.sinkserver.port.range: 28444-28943
      ```
    - Subnet
      ```
      nettyconnector.sinkserver.subnet: 10.162.222.123/24
      ```

      NOTE

      The **nettyconnector.sinkserver.subnet** parameter is set to the subnet (service IP address) of the Flink client by default. If the client and TaskManager are not in the same subnet, an error may occur. Therefore, you need to manually set this parameter to the subnet (service IP address) of TaskManager.

  - NettyServerHandler

    The handler enables the interaction between NettySink and subscribers. After NettySink receives messages, the handler sends these messages out. To ensure data transmission security, this channel is encrypted using SSL. The **nettyconnector.ssl.enabled** configures whether to enable SSL encryption. The SSL encryption is enabled only when **nettyconnector.ssl.enabled** is set to **true**.

- **NettySource**

  NettySource consists of the following major modules:

  - RichParallelSourceFunction

    NettySource inherits RichParallelSinkFunction and attributes of Source operators. The RichParallelSourceFunction API implements following functions:

    - Starts the NettySink operator.
    - Runs the NettySink operator, receives data from subscribers, and injects the data to jobs.
    - Cancels the running of Source operators.

    Following information can be obtained using the attribute of RichParallelSourceFunction:

    - subtaskIndex about the concurrency of each NettySource operator.
    - Concurrency of the NettySource operator.

    When the NettySource operator enters the running stage, the NettyClient status is monitored. Once abnormality occurs, NettyClient is restarted and reconnected to NettyServer, preventing data confusion.

  - RegisterServerHandler

    RegisterServerHandler of NettySource has similar function as the RegisterServerHandler of NettySink. It obtains the IP address, port number, and information of concurrent operators of each subscribed job obtained in the NettySource operator.

- NettyClient

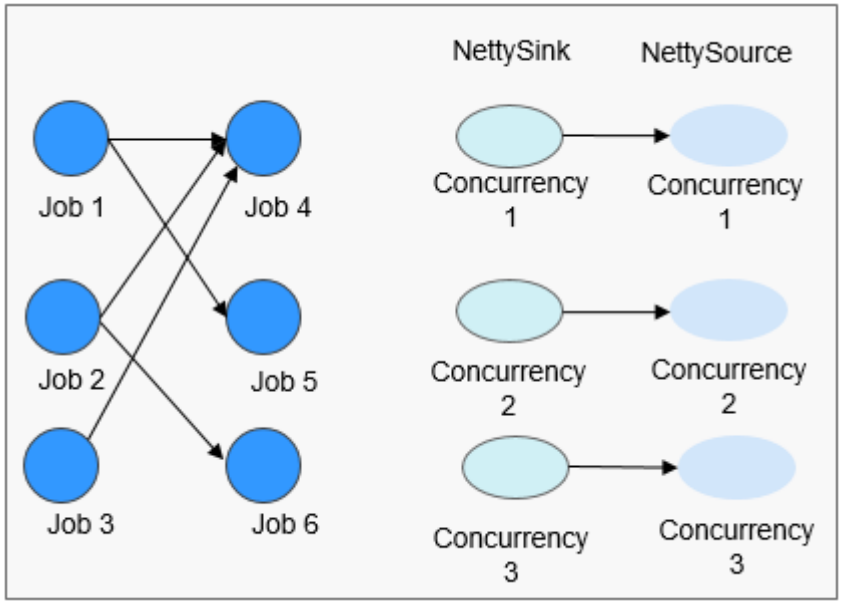
  NettyClient establishes a connection with NettyServer and uses NettyClientHandler to receive data. Each NettySource operator must have a unique name (specified by the user). NettyServer determines whether each client comes from different NettySources based on unique names. When a connection is established between NettyClient and NettyServer, NettyClient is registered with NettyServer and the NettySource name of NettyClient is transferred to NettyServer.

- NettyClientHandler

  The NettyClientHandler enables the interaction with publishers and other operators of the job. When messages are received, NettyClientHandler transfers these messages to the job. To ensure secure data transmission, SSL encryption is enabled for the communication with NettySink. The SSL encryption is enabled only when SSL is enabled and **nettyconnector.ssl.enabled** is set to **true**.

The relationship between the jobs may be many-to-many. The concurrency between each NettySink and NettySource operator is one-to-many, as shown in **Figure 1-25**.

Figure 1-25 Relationship diagram



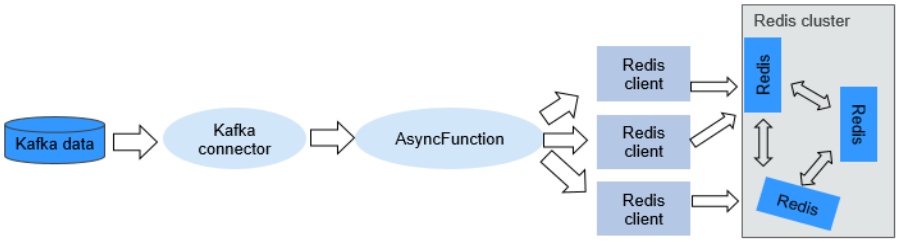
1.4.5.4.3 Configuration Table

Enhanced Open Source Feature: Configuration Table

In some scenarios, users have fixed configuration tables that store basic information. After Flink receives stream data, Flink needs to be configured to match configuration tables. Redis is recommended for storage because the configuration table may be of large size. Redis is a high-performance key-value database with low query latency for stream data.

The detailed process is as follows:

Figure 1-26 Process flow



Data Stored on Redis

Redis is a data structure server supporting various types of values, in addition to key-value storage. The following data types are supported:

- Binary-safe string.
- List: A collection of string elements sorted by their insertion order. It is basically a linked list.
- Sets: Disordered collection of character string elements without repetition.
- Sorted sets: Each string element is associated with a score floating number value. Elements are sorted by score and can be searched.
- Hashes: The map that consists of fields and related values. Fields and values are strings.
- Bit arrays: You can process strings as a series of bits by running certain commands. For example, you are allowed to configure and clear certain bits, calculate the number of bits that are configured to 1, and find the first bit that is configured to 1 or 0.
- HyperLogLogs: A probabilistic data structure which is used to estimate the cardinality of a set.

Redis clusters are used to store configuration tables containing a maximum of 500 million pieces of data, enabling quick query response. Asynchronous I/Os of streams are used to query messages, improving throughput of the data processing.

NOTE

- Redis cluster: In a Redis cluster, Redis is deployed on all nodes in the cluster and data is stored on all nodes with high storage capacity. MRS provides Redis.
- Asynchronous I/O: Asynchronous I/O is used to processes data with maximized data processing throughput, improving the processing efficiency.

Operations on Redis are as follows:

1. Install Redis.

   When installing clusters, you can select Redis provided by MRS.

2. Import configuration tables to Redis.

   You are allowed to select the main key or multiple key columns as the keys based on the feature of the configuration table. If to-be-stored configuration tables contain a large number of attributes, you are advised to storage them in the Hashes data format.

   The Redis component provided by MRS provides the Redis client for inserting queries. For details, see Redis sample code.

NOTE

For details about Redis data types, visit the official website at **https://redis.io/topics/data-types-intro**.

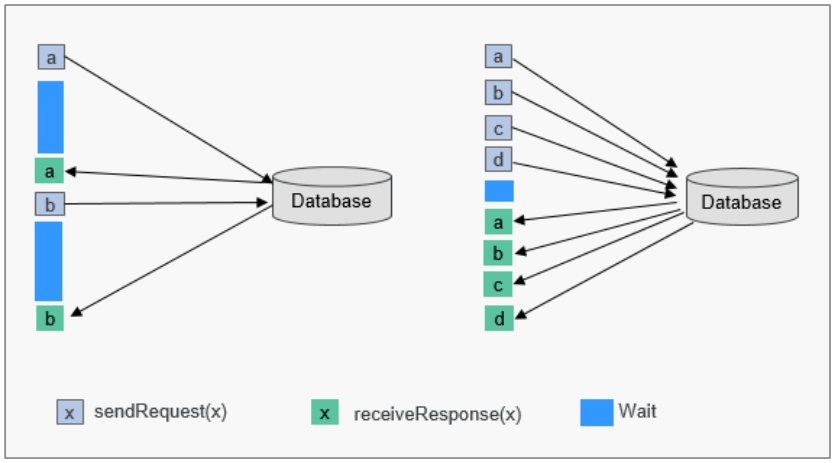
Asynchronous I/Os

When Flink interacts with external systems, such as external databases, the waiting time for responses is too long, reducing data processing efficiency. In asynchronous I/O mode, other requests can be sent without waiting for the response to the previous request, improving data throughput.

The following requirements are required for achieving the API of asynchronous I/O:

- You need to rewrite the **asyncInvoke** method of the AsyncFunction function to implement asynchronous data processing.
- Callback function obtains operator results and AsyncCollector collects the obtained results.

Figure 1-27 Comparison of Async.I/O



- You need to configure the timeout period and maximum capacity.

  Timeout period defines the maximum allowed period for an asynchronous request. The maximum capacity refers to the maximum concurrent number of asynchronous requests. You are advised to configure maximum capacity based on data source features, because an improperly large value will cause high resources consumption and an improperly small value will reduce the throughput.

1.4.5.4.4 Stream SQL Join

Enhanced Open Source Feature: Stream SQL Join

Flink's Table API&SQL is an integrated query API for Scala and Java that allows the composition of queries from relational operators such as selection, filter, and join in an intuitive way. For details about Table API&SQL, visit the official website at **https://ci.apache.org/projects/flink/flink-docs-release-1.12/dev/table/index.html**.

Introduction to Stream SQL Join

SQL Join is used to query data based on the relationship between columns in two or more tables. Flink Stream SQL Join allows you to join two streaming tables and query results from them. Queries similar to the following are supported:

```
SELECT o.proctime, o.productId, o.orderId, s.proctime AS shipTime
FROM Orders AS o
JOIN Shipments AS s
ON o.orderId = s.orderId
AND o.proctime BETWEEN s.proctime AND s.proctime + INTERVAL '1' HOUR;
```

Currently, Stream SQL Join needs to be performed within a specified window. The join operation for data within the window requires at least one equi-join predicate and a join condition that bounds the time on both sides. Such a condition can be defined by two appropriate range predicates (**<**, **<=**, **>=**, **>**), a **BETWEEN** predicate, or a single equality predicate that compares the same type of time attributes (such as processing time or event time) of both input tables.

The following example will join all orders with their corresponding shipments if the order was shipped four hours after the order was received.

```
SELECT *
FROM Orders o, Shipments s
WHERE o.id = s.orderId AND
o.ordertime BETWEEN s.shiptime - INTERVAL '4' HOUR AND s.shiptime
```

NOTE

1. Stream SQL Join supports only inner join.
2. The **ON** clause should include an equal join condition.
3. Time attributes support only the processing time and event time.
4. The window condition supports only the bounded time range, for example, **o.proctime BETWEEN s.proctime - INTERVAL '1' HOUR AND s.proctime + INTERVAL '1' HOUR**. The unbounded range such as **o. proctime > s.proctime** is not supported. The **proctime** attribute of two streams must be included. **o.proctime BETWEEN proctime () AND proctime () + 1** is not supported.

1.4.5.4.5 Flink CEP in SQL

Flink CEP in SQL

Flink allows users to represent complex event processing (CEP) query results in SQL for pattern matching and evaluate event streams on Flink engines.

SQL Query Syntax

CEP SQL is implemented through the **MATCH_RECOGNIZE** SQL syntax. The **MATCH_RECOGNIZE** clause is supported by Oracle SQL since Oracle Database 12c and is used to indicate event pattern matching in SQL. Apache Calcite also supports the **MATCH_RECOGNIZE** clause.

Flink uses Calcite to analyze SQL query results. Therefore, this operation complies with the Apache Calcite syntax.

```
MATCH_RECOGNIZE (
    [ PARTITION BY expression [, expression ]* ]
    [ ORDER BY orderItem [, orderItem ]* ]
    [ MEASURES measureColumn [, measureColumn ]* ]
    [ ONE ROW PER MATCH | ALL ROWS PER MATCH ]
```

```
[ AFTER MATCH
      ( SKIP TO NEXT ROW
      | SKIP PAST LAST ROW
      | SKIP TO FIRST variable
      | SKIP TO LAST variable
      | SKIP TO variable )
]
PATTERN ( pattern )
[ WITHIN intervalLiteral ]
[ SUBSET subsetItem [, subsetItem ]* ]
DEFINE variable AS condition [, variable AS condition ]*
)
```

The syntax elements of the **MATCH_RECOGNIZE** clause are defined as follows:

(Optional) **-PARTITION BY**: defines partition columns. This clause is optional. If this parameter is not defined, the parallelism 1 is used.

(Optional) **-ORDER BY**: defines the sequence of events in a data flow. The **ORDER BY** clause is optional. If it is ignored, non-deterministic sorting is used. Since the order of events is important in pattern matching, this clause should be specified in most cases.

(Optional) **-MEASURES**: specifies the attribute value of the successfully matched event.

(Optional) **-ONE ROW PER MATCH | ALL ROWS PER MATCH**: defines how to output the result. **ONE ROW PER MATCH** indicates that only one row is output for each matching. **ALL ROWS PER MATCH** indicates that one row is output for each matching event.

(Optional) **-AFTER MATCH**: specifies the start position for processing after the next pattern is successfully matched.

-PATTERN: defines the matching pattern as a regular expression. The following operators can be used in the **PATTERN** clause: join operators, quantifier operators (*, +, ?, {n}, {n,}, {n,m}, and {,m}), branch operators (vertical bar |), and differential operators ('{- -}').

(Optional) **-WITHIN**: outputs a pattern clause match only when the match occurs within the specified time.

(Optional) **-SUBSET**: combines one or more associated variables defined in the **DEFINE** clause.

-DEFINE: specifies the Boolean condition, which defines the variables used in the **PATTERN** clause.

In addition, the **MATCH_RECOGNIZE** clause supports the following functions:

-MATCH_NUMBER(): Used in the **MEASURES** clause to allocate the same number to each row that is successfully matched.

-CLASSIFIER(): Used in the **MEASURES** clause to indicate the mapping between matched rows and variables.

-FIRST() and **LAST()**: Used in the **MEASURES** clause to return the value of the expression evaluated in the first or last row of the row set mapped to the schema variable.

-NEXT() and **PREV()**: Used in the **DEFINE** clause to evaluate an expression using the previous or next row in a partition.

-RUNNING and **FINAL** keywords: Used to determine the semantics required for aggregation. **RUNNING** can be used in the **MEASURES** and **DEFINE** clauses, whereas **FINAL** can be used only in the **MEASURES** clause.

- Aggregate functions (**COUNT**, **SUM**, **AVG**, **MAX**, **MIN**): Used in the **MEASURES** and **DEFINE** clauses.

Query Example

The following query finds the V-shaped pattern in the stock price data flow.

```
SELECT *
   FROM MyTable
   MATCH_RECOGNIZE (
     ORDER BY rowtime
     MEASURES
       STRT.name as s_name,
       LAST(DOWN.name) as down_name,
       LAST(UP.name) as up_name
     ONE ROW PER MATCH
     PATTERN (STRT DOWN+ UP+)
     DEFINE
       DOWN AS DOWN.v < PREV(DOWN.v),
       UP AS UP.v > PREV(UP.v)
   )
```

In the following query, the aggregate function **AVG** is used in the **MEASURES** clause of **SUBSET E** consisting of variables related to A and C.

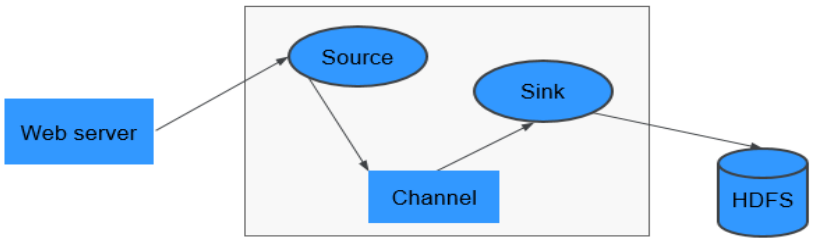
```
SELECT *
   FROM Ticker
   MATCH_RECOGNIZE (
     MEASURES
       AVG(E.price) AS avgPrice
     ONE ROW PER MATCH
     AFTER MATCH SKIP PAST LAST ROW
     PATTERN (A B+ C)
     SUBSET E = (A,C)
     DEFINE
       A AS A.price < 30,
       B AS B.price < 20,
       C AS C.price < 30
   )
```

1.4.6 Flume

1.4.6.1 Flume Basic Principles

Flume is a distributed, reliable, and HA system that supports massive log collection, aggregation, and transmission. Flume supports customization of various data senders in the log system for data collection. In addition, Flume can roughly process data and write data to various data receivers (customizable). A Flume-NG is a branch of Flume. It is simple, small, and easy to deploy. The following figure shows the basic architecture of the Flume-NG.

Figure 1-28 Flume-NG architecture



A Flume-NG consists of agents. Each agent consists of three components (source, channel, and sink). A source is used for receiving data. A channel is used for transmitting data. A sink is used for sending data to the next end.

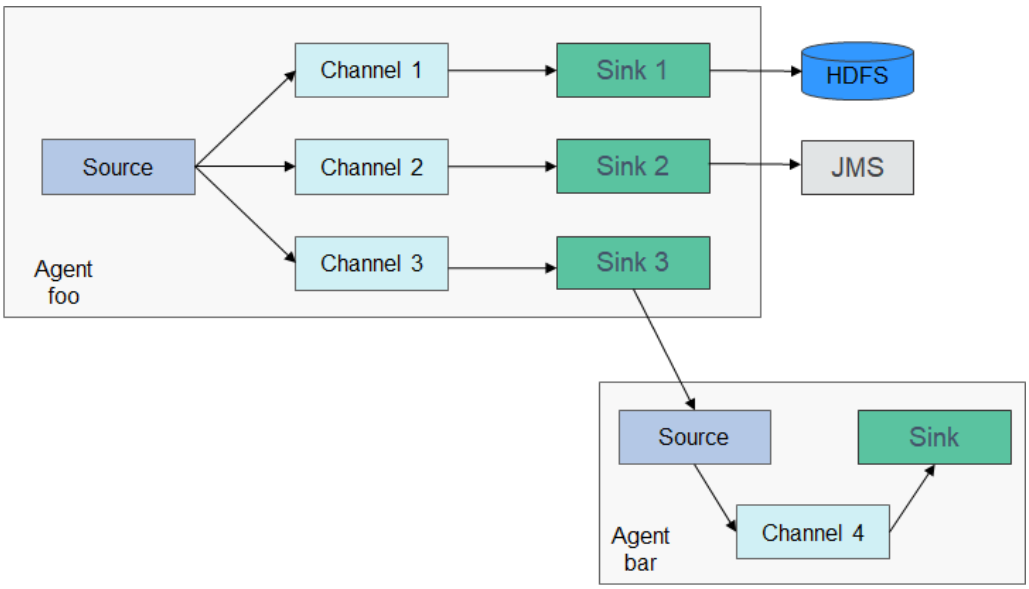
Table 1-4 Module description

Module	Description
Source	A source receives data or generates data by using a special mechanism, and places the data in batches in one or more channels. The source can work in data-driven or polling mode. Typical source types are as follows: ● Sources that are integrated with the system, such as Syslog and Netcat ● Sources that automatically generate events, such as Exec and SEQ ● IPC sources that are used for communication between agents, such as Avro A source must be associated with at least one channel.
Channel	A channel is used to buffer data between a source and a sink. The channel caches data from the source and deletes that data after the sink sends the data to the next-hop channel or final destination. Different channels provide different persistence levels. ● Memory channel: non-persistency ● File channel: Write-Ahead Logging (WAL)-based persistence ● JDBC channel: persistency implemented based on the embedded database The channel supports the transaction feature to ensure simple sequential operations. A channel can work with sources and sinks of any quantity.

Module	Description
Sink	A sink sends data to the next-hop channel or final destination. Once completed, the transmitted data is removed from the channel. Typical sink types are as follows: • Sinks that send storage data to the final destination, such as HDFS and HBase • Sinks that are consumed automatically, such as Null Sink • IPC sinks used for communication between Agents, such as Avro A sink must be associated with a specific channel.

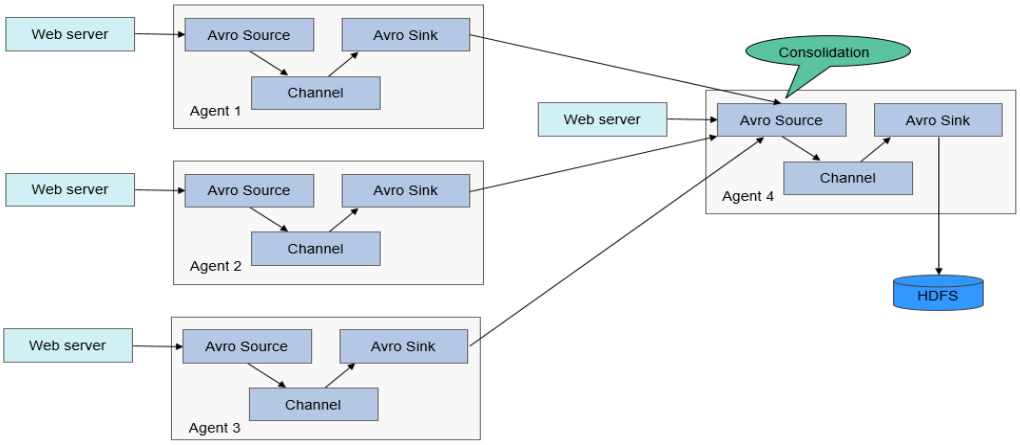
As shown in **Figure 1-29**, a Flume client can have multiple sources, channels, and sinks.

Figure 1-29 Flume structure



The reliability of Flume depends on transaction switchovers between agents. If the next agent breaks down, the channel stores data persistently and transmits data until the agent recovers. The availability of Flume depends on the built-in load balancing and failover mechanisms. Both the channel and agent can be configured with multiple entities between which they can use load balancing policies. Each agent is a Java Virtual Machine (JVM) process. A server can have multiple agents. Collection nodes (for example, Agents 1, 2, 3) process logs. Aggregation nodes (for example, Agent 4) write the logs into HDFS. The agent of each collection node can select multiple aggregation nodes for load balancing.

Figure 1-30 Flume cascading



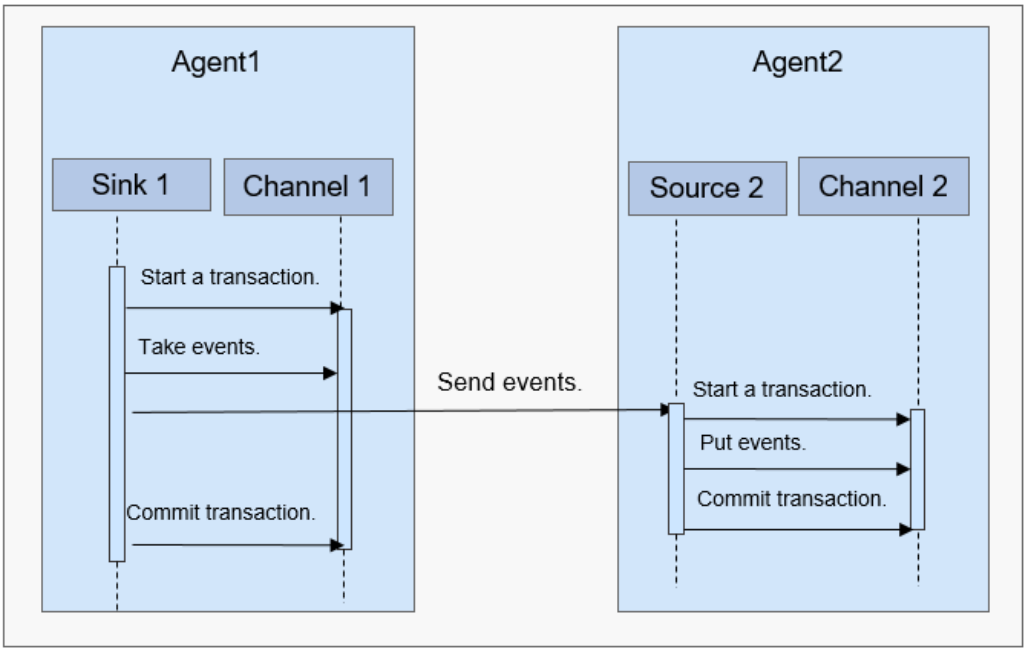
For details about Flume architecture and principles, see **https://flume.apache.org/releases/1.9.0.html**.

Principle

Reliability Between Agents

Figure 1-31 shows the data exchange between agents.

Figure 1-31 Data transmission process



1. Flume ensures reliable data transmission based on transactions. When data flows from one agent to another agent, the two transactions take effect. The sink of Agent 1 (agent that sends a message) needs to obtain a message from a channel and sends the message to Agent 2 (agent that receives the

message). If Agent 2 receives and successfully processes the message, Agent 1 will submit a transaction, indicating a successful and reliable data transmission.

2. When Agent 2 receives the message sent by Agent 1 and starts a new transaction, after the data is processed successfully (written to a channel), Agent 2 submits the transaction and sends a success response to Agent 1.
3. Before a commit operation, if the data transmission fails, the last transcription starts and retransmits the data that fails to be transmitted last time. The commit operation has written the transaction into a disk. Therefore, the last transaction can continue after the process fails and restores.

1.4.6.2 Relationship Between Flume and Other Components

Relationship Between Flume and HDFS

If HDFS is configured as the Flume sink, HDFS functions as the final data storage system of Flume. Flume installs, configures, and writes all transmitted data into HDFS.

Relationship Between Flume and HBase

If HBase is configured as the Flume sink, HBase functions as the final data storage system of Flume. Flume writes all transmitted data into HBase based on configurations.

1.4.6.3 Flume Enhanced Open Source Features

Flume Enhanced Open Source Features

- Improving transmission speed: Multiple lines instead of only one line of data can be specified as an event. This improves the efficiency of code execution and reduces the times of disk writes.
- Transferring ultra-large binary files: According to the current memory usage, Flume automatically adjusts the memory used for transferring ultra-large binary files to prevent out-of-memory.
- Supporting the customization of preparations before and after transmission: Flume supports customized scripts to be run before or after transmission for making preparations.
- Managing client alarms: Flume receives Flume client alarms through MonitorServer and reports the alarms to the alarm management center on MRS Manager.

1.4.7 HBase

1.4.7.1 HBase Basic Principles

HBase undertakes data storage. HBase is an open source, column-oriented, distributed storage system that is suitable for storing massive amounts of unstructured or semi-structured data. It features high reliability, high performance,

and flexible scalability, and supports real-time data read/write. For more information about HBase, see **https://hbase.apache.org/**.

Typical features of a table stored in HBase are as follows:

- Big table (BigTable): One table contains hundred millions of lines and millions of columns.
- Column-oriented: Column-oriented storage, retrieval, and permission control
- Sparse: Null columns in the table do not occupy any storage space.

The HBase component of MRS separates computing from storage. Data can be stored in cloud storage services at low cost, for example, Object Storage Service (OBS), and can be backed up across AZs. MRS supports secondary indexes for HBase and allows adding indexes for column values to filter data by column through native HBase APIs.

HBase architecture

An HBase cluster consists of active and standby HMaster processes and multiple RegionServer processes, as shown in **Figure 1-32**.

Figure 1-32 HBase architecture

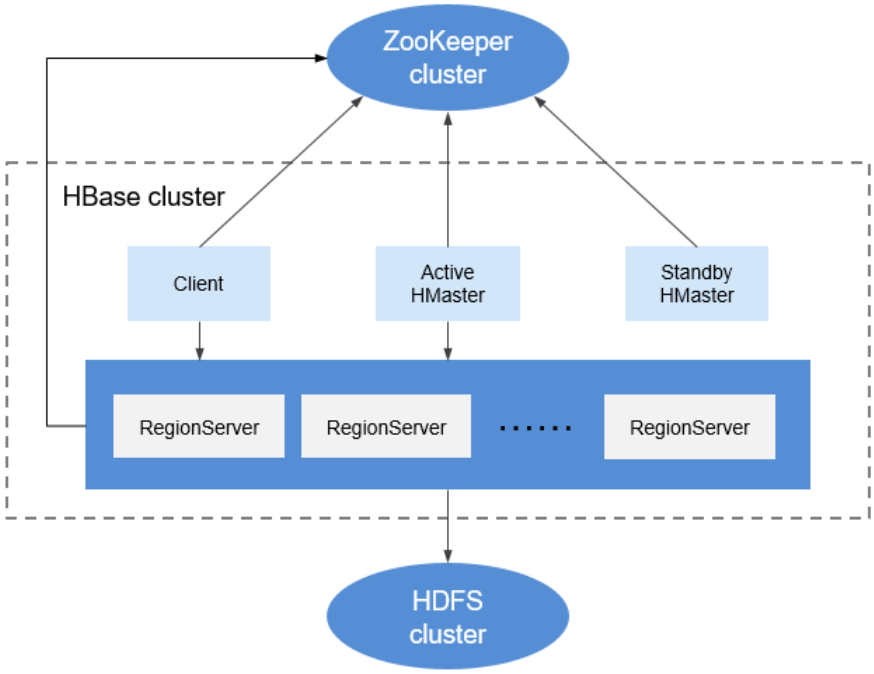


Table 1-5 Module description

Module	Description
Master	Master is also called HMaster. In HA mode, HMaster consists of an active HMaster and a standby HMaster. • Active Master: manages RegionServer in HBase, including the creation, deletion, modification, and query of a table, balances the load of RegionServer, adjusts the distribution of Region, splits Region and distributes Region after it is split, and migrates Region after RegionServer expires. • Standby Master: takes over services when the active HMaster is faulty. The original active HMaster demotes to the standby HMaster after the fault is rectified.
Client	Client communicates with Master for management and with RegionServer for data protection by using the Remote Procedure Call (RPC) mechanism of HBase.
RegionServer	RegionServer provides read and write services of table data as a data processing and computing unit in HBase. RegionServer is deployed with DataNodes of HDFS clusters to store data.
ZooKeeper cluster	ZooKeeper provides distributed coordination services for processes in HBase clusters. Each RegionServer is registered with ZooKeeper so that the active Master can obtain the health status of each RegionServer.
HDFS cluster	HDFS provides highly reliable file storage services for HBase. All HBase data is stored in the HDFS.

HBase Principles

- **HBase Data Model**

  HBase stores data in tables, as shown in **Figure 1-33**. Data in a table is divided into multiple Regions, which are allocated by Master to RegionServers for management.

  Each Region contains data within a RowKey range. An HBase data table contains only one Region at first. As the number of data increases and reaches the upper limit of the Region capacity, the Region is split into two Regions. You can define the RowKey range of a Region when creating a table or define the Region size in the configuration file.

Figure 1-33 HBase data model

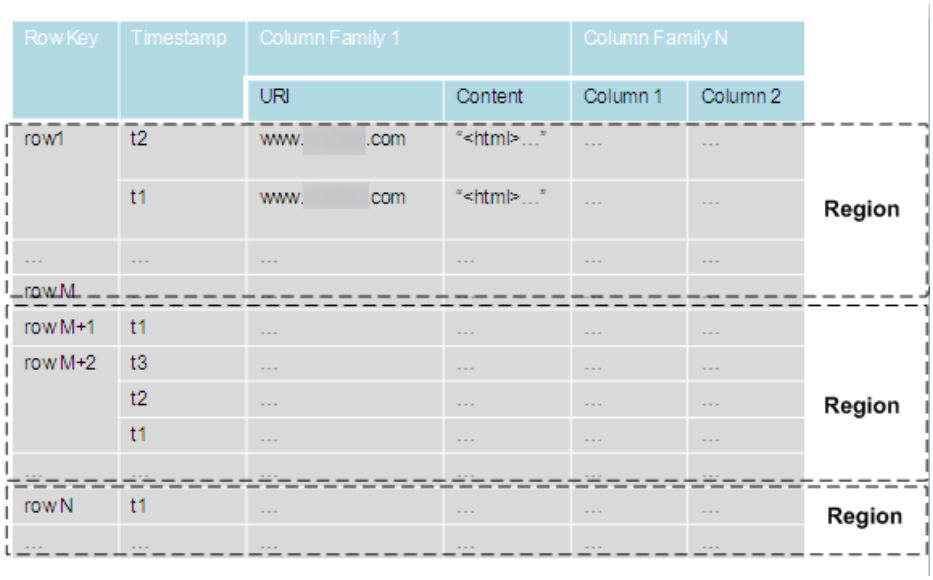


Table 1-6 Concepts

Module	Description
RowKey	Similar to the primary key in a relationship table, which is the unique ID of the data in each row. A RowKey can be a string, integer, or binary string. All records are stored after being sorted by RowKey.
Timestamp	The timestamp of a data operation. Data can be specified with different versions by time stamp. Data of different versions in each cell is stored by time in descending order.
Cell	Minimum storage unit of HBase, consisting of keys and values. A key consists of six fields, namely row, column family, column qualifier, timestamp, type, and MVCC version. Values are the binary data objects.
Column Family	One or multiple horizontal column families form a table. A column family can consist of multiple random columns. A column is a label under a column family, which can be added as required when data is written. The column family supports dynamic expansion so the number and type of columns do not need to be predefined. Columns of a table in HBase are sparsely distributed. The number and type of columns in different rows can be different. Each column family has the independent time to live (TTL). You can lock the row only. Operations on the row in a column family are the same as those on other rows.
Column	Similar to traditional databases, HBase tables also use columns to store data of the same type.

- **RegionServer Data Storage**

RegionServer manages the regions allocated by HMaster. **Figure 1-34** shows the data storage structure of RegionServer.

Figure 1-34 RegionServer data storage structure

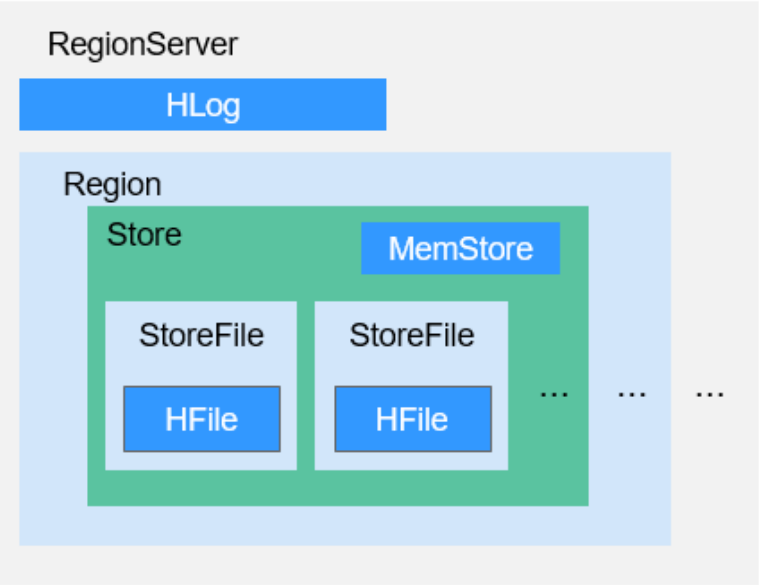


Table 1-7 lists each component of Region described in **Figure 1-34**.

Table 1-7 Region structure description

Module	Description
Store	A Region consists of one or multiple Stores. Each Store maps a column family in **Figure 1-33**.
MemStore	A Store contains one MemStore. The MemStore caches data inserted to a Region by the client. When the MemStore capacity reaches the upper limit, RegionServer flushes data in MemStore to the HDFS.
StoreFile	The data flushed to the HDFS is stored as a StoreFile in the HDFS. As more data is inserted, multiple StoreFiles are generated in a Store. When the number of StoreFiles reaches the upper limit, RegionServer merges multiple StoreFiles into a big StoreFile.
HFile	HFile defines the storage format of StoreFiles in a file system. HFile is the underlying implementation of StoreFile.
HLog	HLogs prevent data loss when RegionServer is faulty. Multiple Regions in a RegionServer share the same HLog.

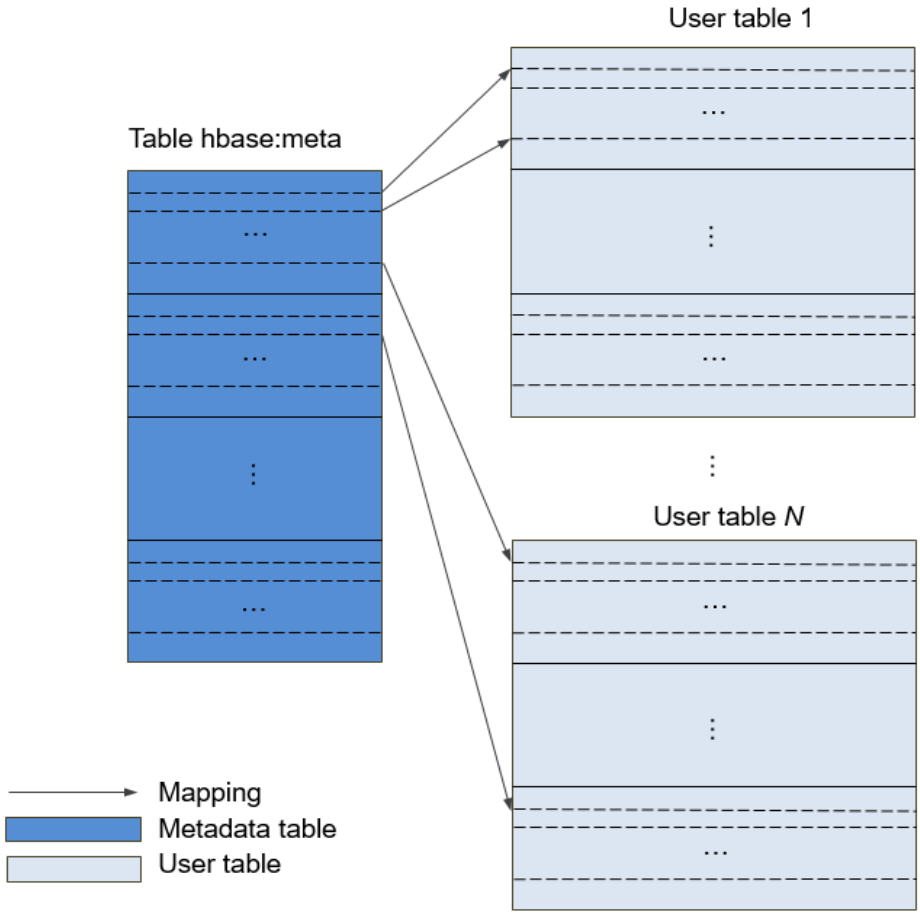
- **Metadata Table**

  The metadata table is a special HBase table, which is used by the client to locate a region. Metadata table includes **hbase:meta** table to record region

information of user tables, such as the region location and start and end RowKey.

Figure 1-35 shows the mapping relationship between metadata tables and user tables.

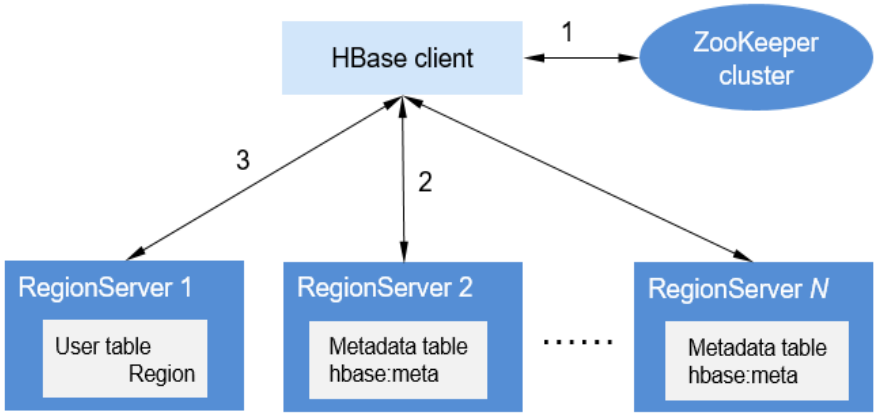
Figure 1-35 Mapping relationships between metadata tables and user tables



- **Data Operation Process**

  Figure 1-36 shows the HBase data operation process.

  Figure 1-36 Data processing



a. When you add, delete, modify, and query HBase data, the HBase client first connects to ZooKeeper to obtain information about the RegionServer where the **hbase:meta** table is located. If you modify the namespace, such as creating and deleting a table, you need to access HMaster to update the meta information.
b. The HBase client connects to the RegionServer where the region of the **hbase:meta** table is located and obtains the RegionServer location where the region of the user table resides.
c. Then the HBase client connects to the RegionServer where the region of the user table is located and issues a data operation command to the RegionServer. The RegionServer executes the command.

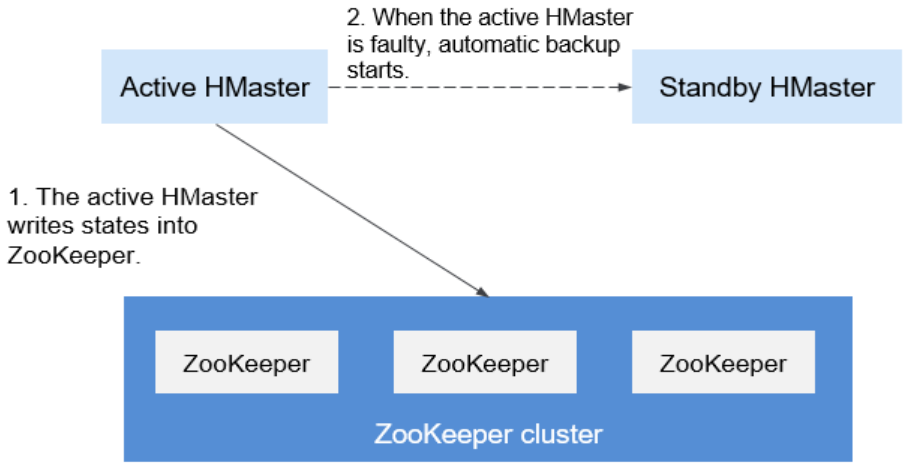
To improve data processing efficiency, the HBase client caches region information of the **hbase:meta** table and user table. When an application initiates a second data operation, the HBase client queries the region information from the memory. If no match is found in the memory, the HBase client performs the preceding operations to obtain region information.

1.4.7.2 HBase HA Solution

HBase HA

HMaster in HBase allocates Regions. When one RegionServer service is stopped, HMaster migrates the corresponding Region to another RegionServer. The HMaster HA feature is brought in to prevent HBase functions from being affected by the HMaster single point of failure (SPOF).

Figure 1-37 HMaster HA implementation architecture



The HMaster HA architecture is implemented by creating the ephemeral ZooKeeper node in a ZooKeeper cluster.

Upon startup, HMaster nodes try to create a master znode in the ZooKeeper cluster. The HMaster node that creates the master znode first becomes the active HMaster, and the other is the standby HMaster.

It will add watch events to the master node. If the service on the active HMaster is stopped, the active HMaster disconnects from the ZooKeeper cluster. After the

session expires, the active HMaster disappears. The standby HMaster detects the disappearance of the active HMaster through watch events and creates a master node to make itself be the active one. Then, the active/standby switchover completes. If the failed node detects existence of the master node after being restarted, it enters the standby state and adds watch events to the master node.

When the client accesses the HBase, it first obtains the HMaster's address based on the master node information on the ZooKeeper and then establishes a connection to the active HMaster.

1.4.7.3 Relationship with Other Components

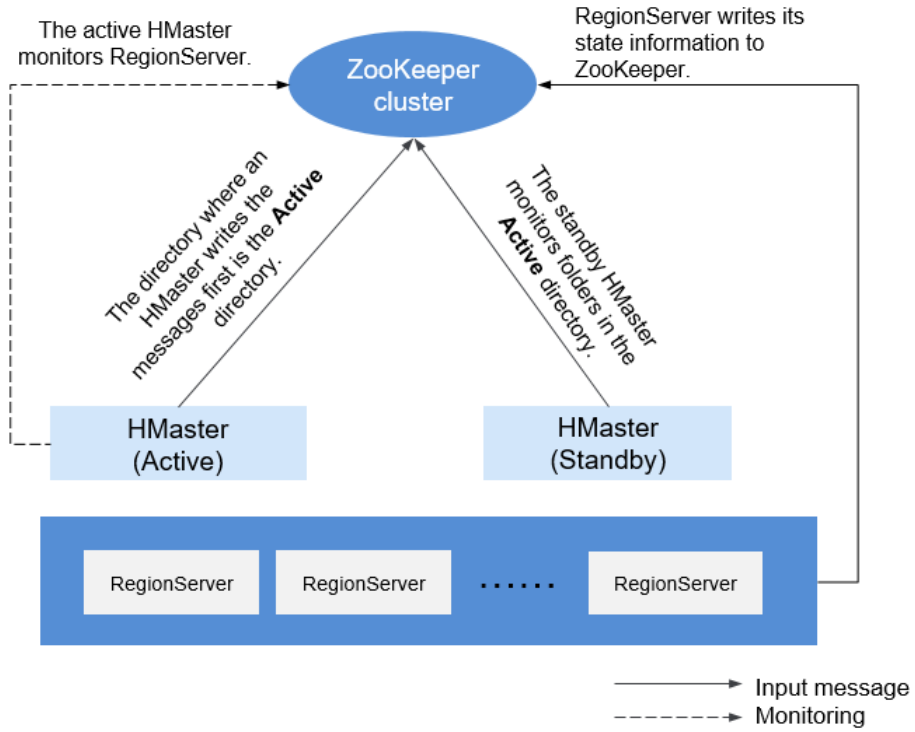
Relationship Between HDFS and HBase

HDFS is the subproject of Apache Hadoop. HBase uses the Hadoop Distributed File System (HDFS) as the file storage system. HBase is located in structured storage layer. The HDFS provides highly reliable support for lower-layer storage of HBase. All the data files of HBase can be stored in the HDFS, except some log files generated by HBase.

Relationship Between ZooKeeper and HBase

Figure 1-38 describes the relationship between ZooKeeper and HBase.

Figure 1-38 Relationship between ZooKeeper and HBase



1. HRegionServer registers itself to ZooKeeper in Ephemeral node. ZooKeeper stores the HBase information, including the HBase metadata and HMaster addresses.
2. HMaster detects the health status of each HRegionServer using ZooKeeper, and monitors them.

3. HBase can deploy multiple HMasters (like HDFS NameNode). When the active HMatser node is faulty, the standby HMaster node obtains the state information of the entire cluster using ZooKeeper, which means that HBase single point faults can be avoided using ZooKeeper.

1.4.7.4 HBase Enhanced Open Source Features

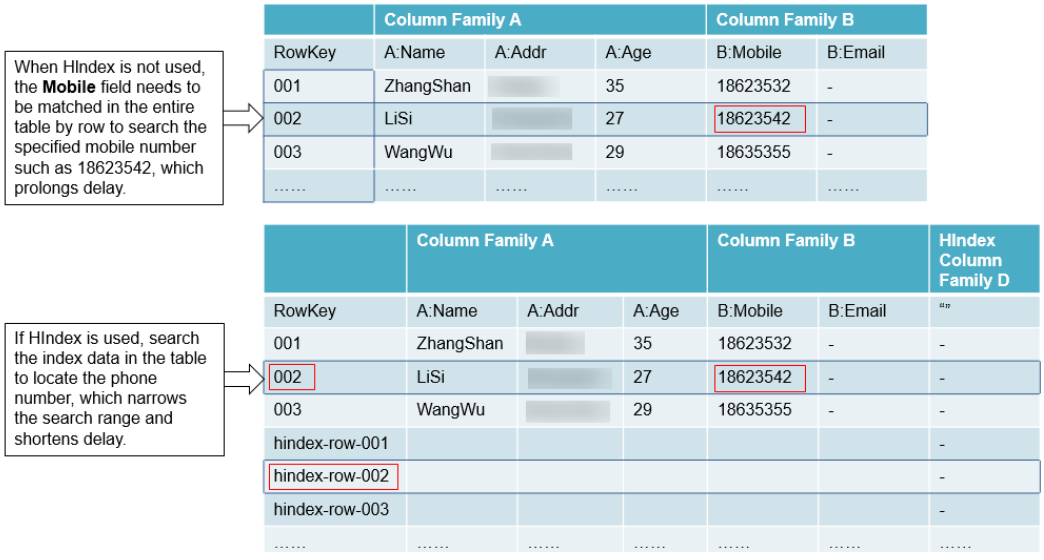
HIndex

HBase is a distributed storage database of the Key-Value type. Data of a table is sorted in the alphabetic order based on row keys. If you query data based on a specified row key or scan data in the scale of a specified row key, HBase can quickly locate the target data, enhancing the efficiency.

However, in most actual scenarios, you need to query the data of which the column value is *XXX*. HBase provides the Filter feature to query data with a specific column value. All data is scanned in the order of row keys, and then the data is matched with the specific column value until the required data is found. The Filter feature scans some unnecessary data to obtain the only required data. Therefore, the Filter feature cannot meet the requirements of frequent queries with high performance standards.

HBase HIndex is designed to address these issues. HBase HIndex enables HBase to query data based on specific column values.

Figure 1-39 HIndex



- Rolling upgrade is not supported for index data.
- Restrictions of combined indexes:
  - All columns involved in combined indexes must be entered or deleted in a single mutation. Otherwise, inconsistency will occur.

    Index: **IDX1=>cf1:[q1->datatype],[q2];cf2:[q2->datatype]**

    Correct write operations:

    ```
    Put put = new Put(Bytes.toBytes("row"));
    put.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q1"), Bytes.toBytes("valueA"));
    ```

```
put.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q2"), Bytes.toBytes("valueB"));
put.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q2"), Bytes.toBytes("valueC"));
table.put(put);
```

Incorrect write operations:

```
Put put1 = new Put(Bytes.toBytes("row"));
put1.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q1"), Bytes.toBytes("valueA"));
table.put(put1);
Put put2 = new Put(Bytes.toBytes("row"));
put2.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q2"), Bytes.toBytes("valueB"));
table.put(put2);
Put put3 = new Put(Bytes.toBytes("row"));
put3.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q2"), Bytes.toBytes("valueC"));
table.put(put3);
```

– The combined conditions-based query is supported only when the combined index column contains filter criteria, or StartRow and StopRow are not specified for some index columns.

  Index: **IDX1=>cf1:[q1->datatype],[q2];cf2:[q1->datatype]**

  Correct query operations:

```
scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',>=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf1','q2',>=,'binary:valueB',true,true) AND
SingleColumnValueFilter('cf2','q1',>=,'binary:valueC',true,true) "}

scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf1','q2',>=,'binary:valueB',true,true)" }

scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',>=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf1','q2',>=,'binary:valueB',true,true) AND
SingleColumnValueFilter('cf2','q1',>=,'binary:valueC',true,true)",STARTROW=>'row001',STOPROW
=>'row100'}
```

  Incorrect query operations:

```
scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',>=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf1','q2',>=,'binary:valueB',true,true) AND
SingleColumnValueFilter('cf2','q1',>=,'binary:valueC',true,true)  AND
SingleColumnValueFilter('cf2','q2',>=,'binary:valueD',true,true)"}

scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf2','q1',>=,'binary:valueC',true,true)" }

scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf2','q2',>=,'binary:valueD',true,true)" }

scan 'table', {FILTER=>"SingleColumnValueFilter('cf1','q1',=,'binary:valueA',true,true) AND
SingleColumnValueFilter('cf1','q2',>=,'binary:valueB',true,true)" ,STARTROW=>'row001',STOPROW
=>'row100' }
```

- Do not explicitly configure any split policy for tables with index data.
- Other mutation operations, such as **increment** and **append**, are not supported.
- Index of the column with **maxVersions** greater than 1 is not supported.
- The data index column in a row cannot be updated.

  Index 1: **IDX1=>cf1:[q1->datatype],[q2];cf2:[q1->datatype]**

  Index 2: **IDX2=>cf2:[q2->datatype]**

  Correct update operations:

```
Put put1 = new Put(Bytes.toBytes("row"));
put1.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q1"), Bytes.toBytes("valueA"));
put1.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q2"), Bytes.toBytes("valueB"));
put1.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q1"), Bytes.toBytes("valueC"));
put1.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q2"), Bytes.toBytes("valueD"));
table.put(put1);
```

```
Put put2 = new Put(Bytes.toBytes("row"));
put2.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q3"), Bytes.toBytes("valueE"));
put2.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q3"), Bytes.toBytes("valueF"));
table.put(put2);
```

Incorrect update operations:

```
Put put1 = new Put(Bytes.toBytes("row"));
put1.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q1"), Bytes.toBytes("valueA"));
put1.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q2"), Bytes.toBytes("valueB"));
put1.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q1"), Bytes.toBytes("valueC"));
put1.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q2"), Bytes.toBytes("valueD"));
table.put(put1);

Put put2 = new Put(Bytes.toBytes("row"));
put2.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q1"), Bytes.toBytes("valueA_new"));
put2.addColumn(Bytes.toBytes("cf1"), Bytes.toBytes("q2"), Bytes.toBytes("valueB_new"));
put2.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q1"), Bytes.toBytes("valueC_new"));
put2.addColumn(Bytes.toBytes("cf2"), Bytes.toBytes("q2"), Bytes.toBytes("valueD_new"));
table.put(put2);
```

- The table to which an index is added cannot contain a value greater than 32 KB.
- If user data is deleted due to the expiration of the column-level TTL, the corresponding index data is not deleted immediately. It will be deleted in the major compaction operation.
- The TTL of the user column family cannot be modified after the index is created.
  - If the TTL of a column family increases after an index is created, delete the index and re-create one. Otherwise, some generated index data will be deleted before user data is deleted.
  - If the TTL value of the column family decreases after an index is created, the index data will be deleted after user data is deleted.
- The index query does not support the reverse operation, and the query results are disordered.
- The index does not support the **clone snapshot** operation.
- The index table must use HIndexWALPlayer to replay logs. WALPlayer cannot be used to replay logs.

```
hbase org.apache.hadoop.hbase.hindex.mapreduce.HIndexWALPlayer
Usage: WALPlayer [options] <wal inputdir> <tables> [<tableMappings>]
Read all WAL entries for <tables>.
If no tables ("") are specific, all tables are imported.
(Careful, even -ROOT- and hbase:meta entries will be imported in that case.)
Otherwise <tables> is a comma separated list of tables.

The WAL entries can be mapped to new set of tables via <tableMapping>.
<tableMapping> is a command separated list of targettables.
If specified, each table in <tables> must have a mapping.

By default WALPlayer will load data directly into HBase.
To generate HFiles for a bulk data load instead, pass the option:
  -Dwal.bulk.output=/path/for/output
  (Only one table can be specified, and no mapping is allowed!)
Other options: (specify time range to WAL edit to consider)
  -Dwal.start.time=[date|ms]
  -Dwal.end.time=[date|ms]
For performance also consider the following options:
  -Dmapreduce.map.speculative=false
  -Dmapreduce.reduce.speculative=false
```

- When the **deleteall** command is executed for the index table, the performance is low.

- The index table does not support HBCK. To use HBCK to repair the index table, delete the index data first.

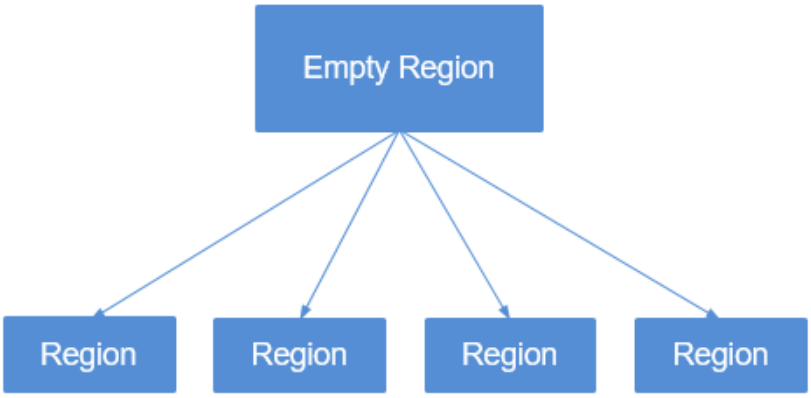
Multi-point Division

When you create tables that are pre-divided by region in HBase, you may not know the data distribution trend so the division by region may be inappropriate. After the system runs for a period, regions need to be divided again to achieve better performance. Only empty regions can be divided.

The region division function delivered with HBase divides regions only when they reach the threshold. This is called "single point division".

To achieve better performance when regions are divided based on user requirements, multi-point division is developed, which is also called "dynamic division". That is, an empty region is pre-divided into multiple regions to prevent performance deterioration caused by insufficient region space.

Figure 1-40 Multi-point division



Connection Limitation

Too many sessions mean that too many queries and MapReduce tasks are running on HBase, which compromises HBase performance and even causes service rejection. You can configure parameters to limit the maximum number of sessions that can be established between the client and the HBase server to achieve HBase overload protection.

Improved Disaster Recovery

The disaster recovery (DR) capabilities between the active and standby clusters can enhance HA of the HBase data. The active cluster provides data services and the standby cluster backs up data. If the active cluster is faulty, the standby cluster takes over data services. Compared with the open source replication function, this function is enhanced as follows:

1. The standby cluster whitelist function is only applicable to pushing data to a specified cluster IP address.
2. In the open source version, replication is synchronized based on WAL, and data backup is implemented by replaying WAL in the standby cluster. For

BulkLoad operations, since no WAL is generated, data will not be replicated to the standby cluster. By recording BulkLoad operations on the WAL and synchronizing them to the standby cluster, the standby cluster can read BulkLoad operation records through WAL and load HFile in the active cluster to the standby cluster to implement data backup.

3. In the open source version, HBase filters ACLs. Therefore, ACL information will not be synchronized to the standby cluster. By adding a filter (**org.apache.hadoop.hbase.replication.SystemTableWALEntryFilterAllowACL**), ACL information can be synchronized to the standby cluster. You can configure **hbase.replication.filter.sytemWALEntryFilter** to enable the filter and implement ACL synchronization.
4. As for read-only restriction of the standby cluster, only super users within the standby cluster can modify the HBase of the standby cluster. In other words, HBase clients outside the standby cluster can only read the HBase of the standby cluster.

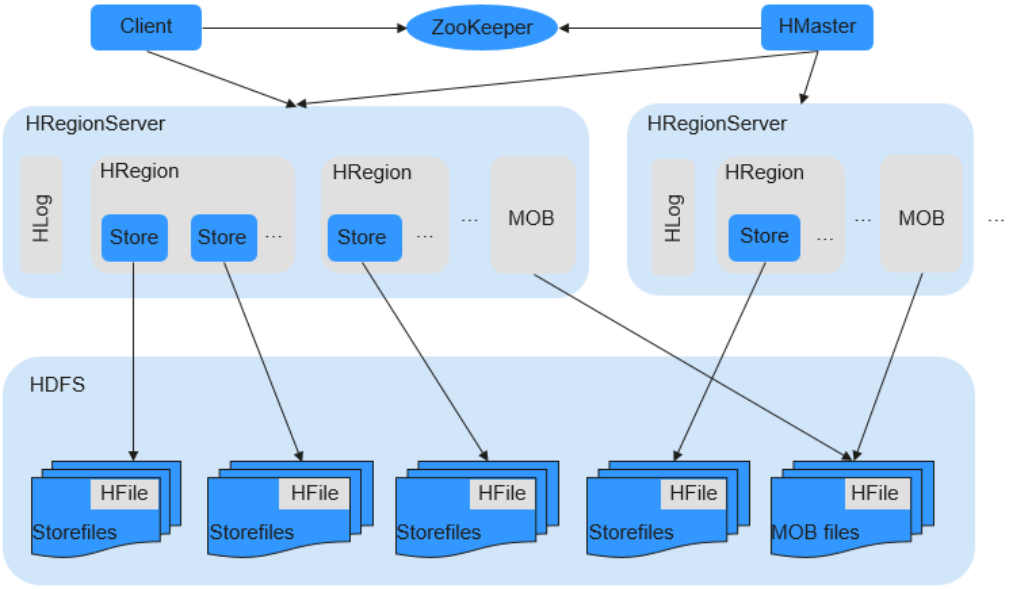
HBase MOB

In the actual application scenarios, data in various sizes needs to be stored, for example, image data and documents. Data whose size is smaller than 10 MB can be stored in HBase. HBase can yield the best read-and-write performance for data whose size is smaller than 100 KB. If the size of data stored in HBase is greater than 100 KB or even reaches 10 MB and the same number of data files are inserted, the total data amount is large, causing frequent compaction and split, high CPU consumption, high disk I/O frequency, and low performance.

MOB data (whose size ranges from 100 KB to 10 MB) is stored in a file system (for example, HDFS) in HFile format. The expiredMobFileCleaner and Sweeper tools are used to manage HFiles and save the address and size information about the HFiles to the store of HBase as values. This greatly decreases the compaction and split frequency in HBase and improves performance.

As shown in **Figure 1-41**, MOB indicates mobstore stored on HRegion. Mobstore stores keys and values. Wherein, a key is the corresponding key in HBase, and a value is the reference address and data offset stored in the file system. When reading data, mobstore uses its own scanner to read key-value data objects and uses the address and data size information in the value to obtain target data from the file system.

Figure 1-41 MOB data storage principle



HFS

HBase FileStream (HFS) is an independent HBase file storage module. It is used in MRS upper-layer applications by encapsulating HBase and HDFS interfaces to provide these upper-layer applications with functions such as file storage, read, and deletion.

In the Hadoop ecosystem, the HDFS and HBase face tough problems in mass file storage in some scenarios:

- If a large number of small files are stored in HDFS, the NameNode will be under great pressure.
- Some large files cannot be directly stored on HBase due to HBase APIs and internal mechanisms.

HFS is developed for the mixed storage of massive small files and some large files in Hadoop. Simply speaking, massive small files (smaller than 10 MB) and some large files (greater than 10 MB) need to be stored in HBase tables.

For such a scenario, HFS provides unified operation APIs similar to HBase function APIs.

Multiple RegionServers Deployed on the Same Server

Multiple RegionServers can be deployed on one node to improve HBase resource utilization.

If only one RegionServer is deployed, resource utilization is low due to the following reasons:

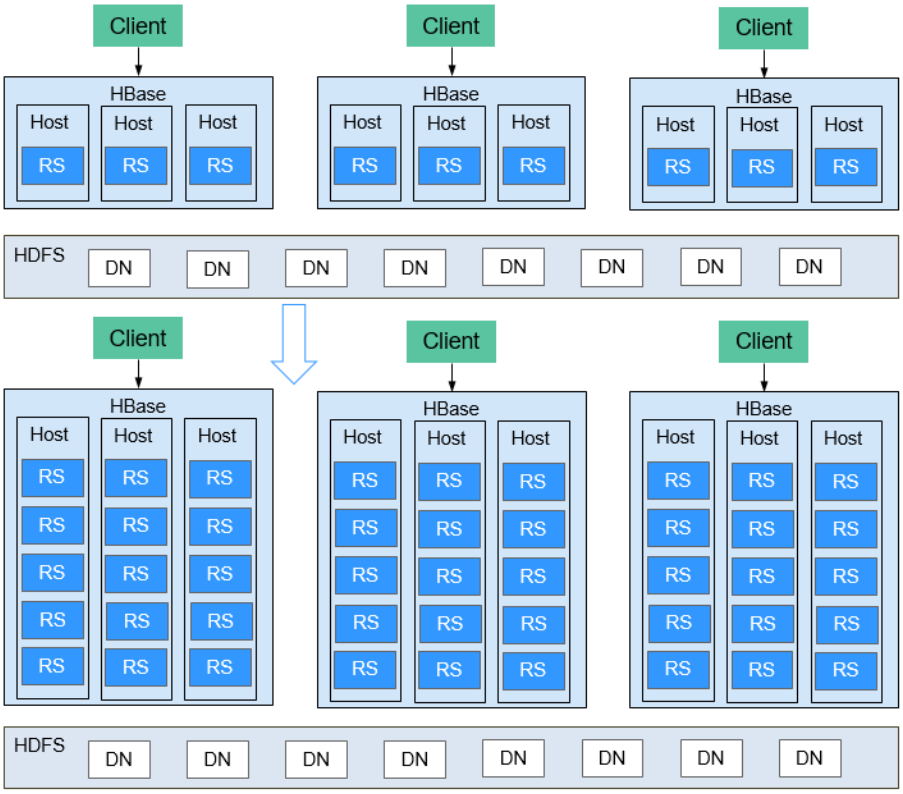
1. A RegionServer supports a limited number of regions, and therefore memory and CPU resources cannot be fully used.
2. A single RegionServer supports a maximum of 20 TB data, of which two copies require 40 TB, and three copies require 60 TB. In this case, 96 TB capacity cannot be used up.

3. Poor write performance: One RegionServer is deployed on a physical server, and only one HLog exists. Only three disks can be written at the same time.

The HBase resource utilization can be improved when multiple RegionServers are deployed on the same server.

1. A physical server can be configured with a maximum of five RegionServers. The number of RegionServers deployed on each physical server can be configured as required.
2. Resources such as memory, disks, and CPUs can be fully used.
3. A physical server supports a maximum of five HLogs and allows data to be written to 15 disks at the same time, significantly improving write performance.

Figure 1-42 Improved HBase resource utilization

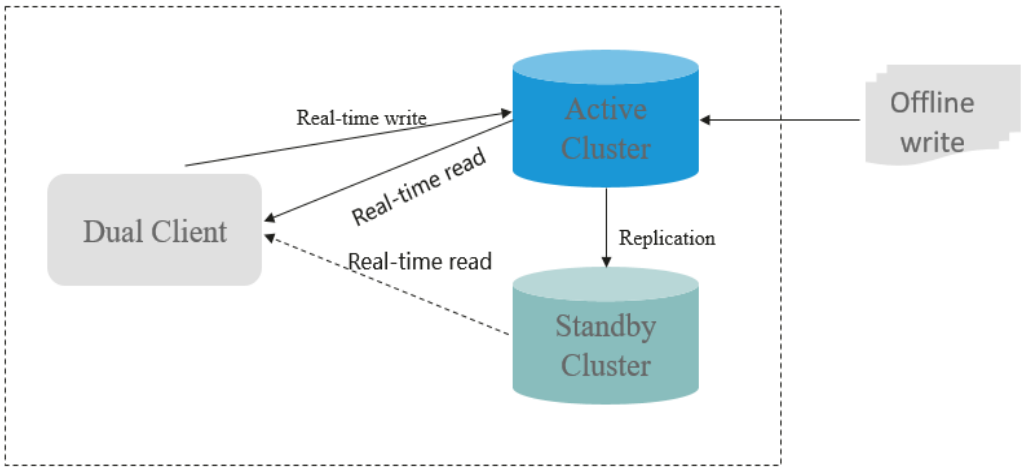


HBase Dual-Read

In the HBase storage scenario, it is difficult to ensure 99.9% query stability due to GC, network jitter, and bad sectors of disks. The HBase dual-read feature is added to meet the requirements of low glitches during large-data-volume random read.

The HBase dual-read feature is based on the DR capability of the active and standby clusters. The probability that the two clusters generate glitches at the same time is far less than that of one cluster. The dual-cluster concurrent access mode is used to ensure query stability. When a user initiates a query request, the HBase service of the two clusters is queried at the same time. If the active cluster does not return any result after a period of time (the maximum tolerable glitch

time), the data of the cluster with the fastest response can be used. The following figure shows the working principle.



1.4.8 HDFS

1.4.8.1 HDFS Basic Principles

Hadoop Distributed File System (HDFS) implements reliable and distributed read/write of massive amounts of data. HDFS is applicable to the scenario where data read/write features "write once and read multiple times". However, the write operation is performed in sequence, that is, it is a write operation performed during file creation or an adding operation performed behind the existing file. HDFS ensures that only one caller can perform write operation on a file but multiple callers can perform read operation on the file at the same time.

Architecture

HDFS consists of active and standby NameNodes and multiple DataNodes, as shown in **Figure 1-43**.

HDFS works in master/slave architecture. NameNodes run on the master (active) node, and DataNodes run on the slave (standby) node. ZKFC should run along with the NameNodes.

The communication between NameNodes and DataNodes is based on Transmission Control Protocol (TCP)/Internet Protocol (IP). The NameNode, DataNode, ZKFC, and JournalNode can be deployed on Linux servers.

Figure 1-43 HA HDFS architecture

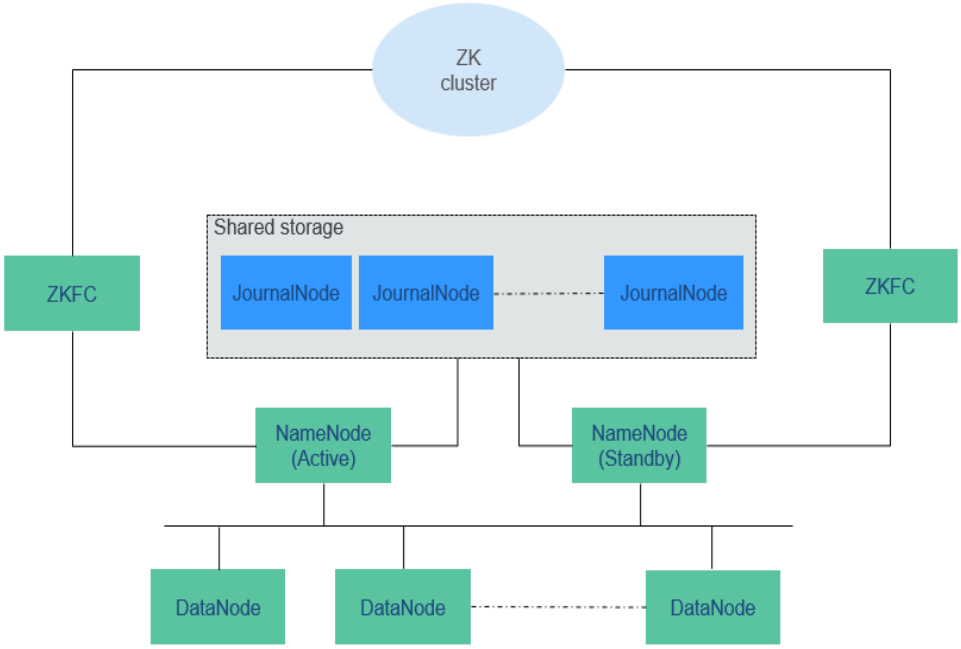


Table 1-8 describes the functions of each module shown in **Figure 1-43**.

Table 1-8 Module description

Module	Description
NameNode	A NameNode is used to manage the namespace, directory structure, and metadata information of a file system and provide the backup mechanism. The NameNode is classified into the following two types: • Active NameNode: manages the namespace, maintains the directory structure and metadata of file systems, and records the mapping relationships between data blocks and files to which the data blocks belong. • Standby NameNode: synchronizes with the data in the active NameNode, and takes over services from the active NameNode when the active NameNode is faulty. • Observer NameNode: synchronizes with the data in the active NameNode, and processes read requests from the client.
DataNode	A DataNode is used to store data blocks of each file and periodically report the storage status to the NameNode.
JournalNode	In HA cluster, synchronizes metadata between the active and standby NameNodes.
ZKFC	ZKFC must be deployed for each NameNode. It monitors NameNode status and writes status information to ZooKeeper. ZKFC also has permissions to select the active NameNode.

Module	Description
ZK Cluster	ZooKeeper is a coordination service which helps the ZKFC to elect the active NameNode.
HttpFS gateway	HttpFS is a single stateless gateway process which provides the WebHDFS REST API for external processes and FileSystem API for the HDFS. HttpFS is used for data transmission between different versions of Hadoop. It is also used as a gateway to access the HDFS behind a firewall.

- **HDFS HA Architecture**

  HA is used to resolve the SPOF problem of NameNode. This feature provides a standby NameNode for the active NameNode. When the active NameNode is faulty, the standby NameNode can quickly take over to continuously provide services for external systems.

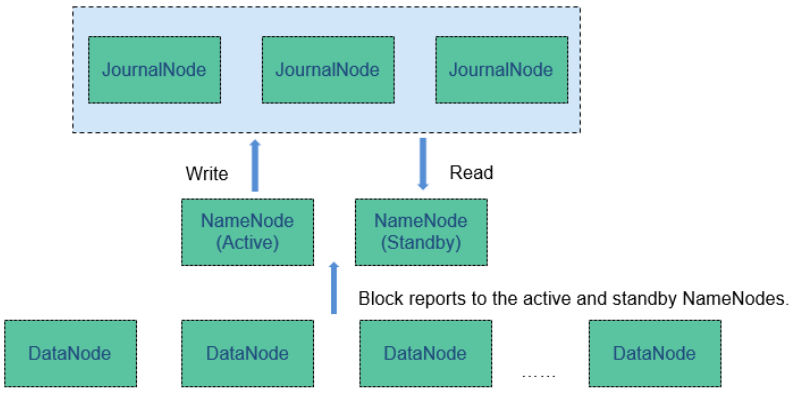
  In a typical HDFS HA scenario, there are usually two NameNodes. One is in the active state, and the other in the standby state.

  A shared storage system is required to support metadata synchronization of the active and standby NameNodes. This version provides Quorum Journal Manager (QJM) HA solution, as shown in **Figure 1-44**. A group of JournalNodes are used to synchronize metadata between the active and standby NameNodes.

  Generally, an odd number (2N+1) of JournalNodes are configured, and at least three JournalNodes are required. For one metadata update message, data writing is considered successful as long as data writing is successful on N +1 JournalNodes. In this case, data writing failure of a maximum of N JournalNodes is allowed. For example, when there are three JournalNodes, data writing failure of one JournalNode is allowed; when there are five JournalNodes, data writing failure of two JournalNodes is allowed.

  JournalNode is a lightweight daemon process and shares a host with other services of Hadoop. It is recommended that the JournalNode be deployed on the control node to prevent data writing failure on the JournalNode during massive data transmission.

  Figure 1-44 QJM-based HDFS architecture

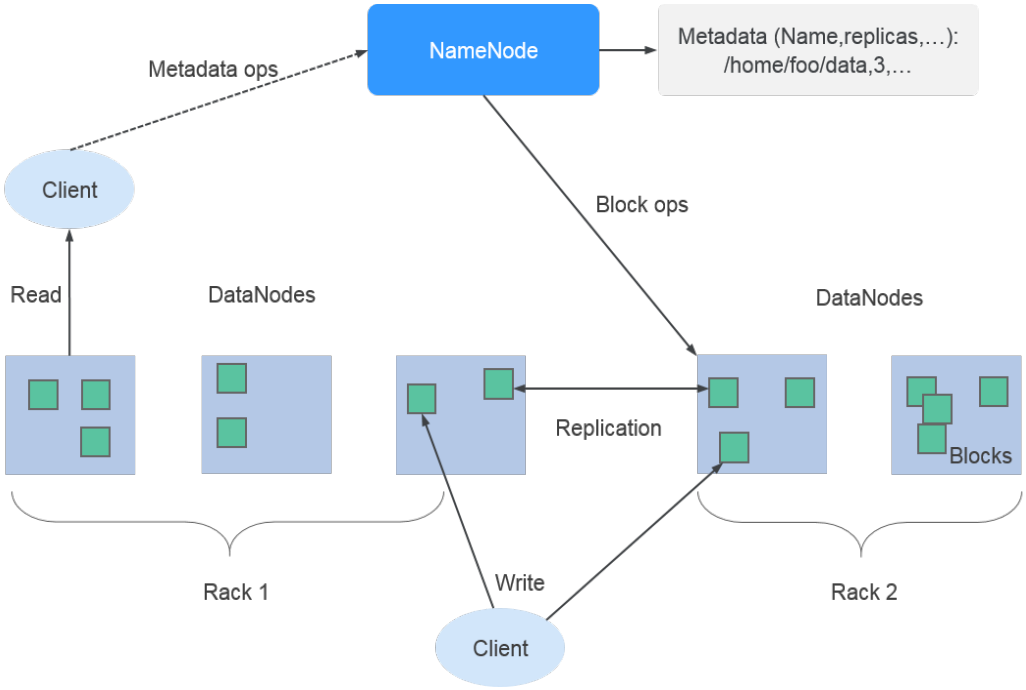


Principle

MRS uses the HDFS copy mechanism to ensure data reliability. One backup file is automatically generated for each file saved in HDFS, that is, two copies are generated in total. The number of HDFS copies can be queried using the **dfs.replication** parameter.

- When the Core node specification of the MRS cluster is set to non-local hard disk drive (HDD) and the cluster has only one Core node, the default number of HDFS copies is 1. If the number of Core nodes in the cluster is greater than or equal to 2, the default number of HDFS copies is 2.
- When the Core node specification of the MRS cluster is set to local disk and the cluster has only one Core node, the default number of HDFS copies is 1. If there are two Core nodes in the cluster, the default number of HDFS copies is 2. If the number of Core nodes in the cluster is greater than or equal to 3, the default number of HDFS copies is 3.

Figure 1-45 HDFS architecture



The HDFS component of MRS supports the following features:

- Supports erasure code, reducing data redundancy to 50% and improving reliability. In addition, the striped block storage structure is introduced to maximize the use of the capability of a single node and multiple disks in an existing cluster. After the coding process is introduced, the data write performance is improved, and the performance is close to that with the multi-copy redundancy.
- Supports balanced node scheduling on HDFS and balanced disk scheduling on a single node, improving HDFS storage performance after node or disk scale-out.

For details about the Hadoop architecture and principles, see **https://hadoop.apache.org/**.

1.4.8.2 HDFS HA Solution

HDFS HA Background

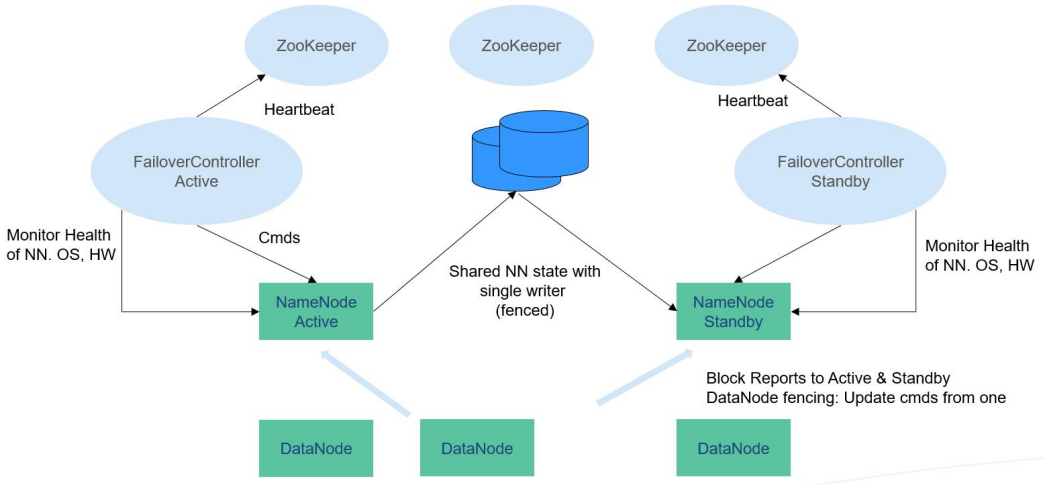
In versions earlier than Hadoop 2.0.0, SPOF occurs in the HDFS cluster. Each cluster has only one NameNode. If the host where the NameNode is located is faulty, the HDFS cluster cannot be used unless the NameNode is restarted or started on another host. This affects the overall availability of HDFS in the following aspects:

1. In the case of an unplanned event such as host breakdown, the cluster would be unavailable until the NameNode is restarted.
2. Planned maintenance tasks, such as software and hardware upgrade, will cause the cluster stop working.

To solve the preceding problems, the HDFS HA solution enables a hot-swap NameNode backup for NameNodes in a cluster in automatic or manual (configurable) mode. When a machine fails (due to hardware failure), the active/standby NameNode switches over automatically in a short time. When the active NameNode needs to be maintained, the MRS cluster administrator can manually perform an active/standby NameNode switchover to ensure cluster availability during maintenance. For details about the automatic failover of HDFS, see **https://hadoop.apache.org/docs/r3.1.1/hadoop-project-dist/hadoop-hdfs/HDFSHighAvailabilityWithQJM.html#Automatic_Failover**.

HDFS HA Implementation

Figure 1-46 Typical HA deployment



In a typical HA cluster (as shown in **Figure 1-46**), two NameNodes need to be configured on two independent servers, respectively. At any time point, one NameNode is in the active state, and the other NameNode is in the standby state. The active NameNode is responsible for all client operations in the cluster, while the standby NameNode maintains synchronization with the active node to provide fast switchover if necessary.

To keep the data synchronized with each other, both nodes communicate with a group of JournalNodes. When the active node modifies any file system's metadata,

it will store the modification log to a majority of these JournalNodes. For example, if there are three JournalNodes, then the log will be saved on two of them at least. The standby node monitors changes of JournalNodes and synchronizes changes from the active node. Based on the modification log, the standby node applies the changes to the metadata of the local file system. Once a switchover occurs, the standby node can ensure its status is the same as that of the active node. This ensures that the metadata of the file system is synchronized between the active and standby nodes if the switchover is incurred by the failure of the active node.

To ensure fast switchover, the standby node needs to have the latest block information. Therefore, DataNodes send block information and heartbeat messages to two NameNodes at the same time.

It is vital for an HA cluster that only one of the NameNodes be active at any time. Otherwise, the namespace state would split into two parts, risking data loss or other incorrect results. To prevent the so-called "split-brain scenario", the JournalNodes will only ever allow a single NameNode to write data to it at a time. During switchover, the NameNode which is to become active will take over the role of writing data to JournalNodes. This effectively prevents the other NameNodes from being in the active state, allowing the new active node to safely proceed with switchover.

For more information about the HDFS HA solution, visit the following website:

http://hadoop.apache.org/docs/r3.1.1/hadoop-project-dist/hadoop-hdfs/HDFSHighAvailabilityWithQJM.html

1.4.8.3 Relationship Between HDFS and Other Components

Relationship Between HDFS and HBase

HDFS is a subproject of Apache Hadoop, which is used as the file storage system for HBase. HBase is located in the structured storage layer. HDFS provides highly reliable support for lower-layer storage of HBase. All the data files of HBase can be stored in the HDFS, except some log files generated by HBase.

Relationship Between HDFS and MapReduce

- HDFS features high fault tolerance and high throughput, and can be deployed on low-cost hardware for storing data of applications with massive data sets.
- MapReduce is a programming model used for parallel computation of large data sets (larger than 1 TB). Data computed by MapReduce comes from multiple data sources, such as Local FileSystem, HDFS, and databases. Most data comes from the HDFS. The high throughput of HDFS can be used to read massive data. After being computed, data can be stored in HDFS.

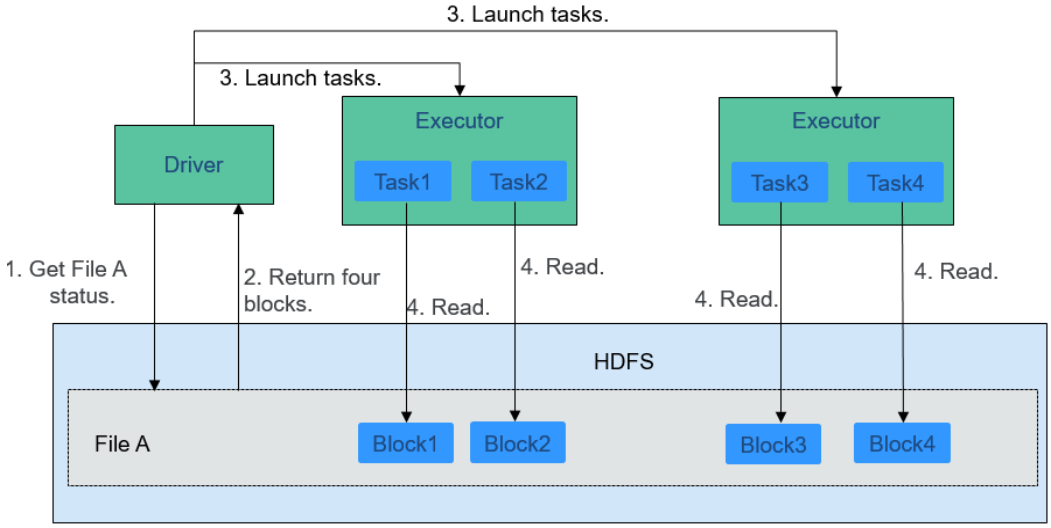
Relationship Between HDFS and Spark

Data computed by Spark comes from multiple data sources, such as local files and HDFS. Most data comes from HDFS which can read data in large scale for parallel computing. After being computed, data can be stored in HDFS.

Spark involves Driver and Executor. Driver schedules tasks and Executor runs tasks.

Figure 1-47 shows how data is read from a file.

Figure 1-47 File reading process

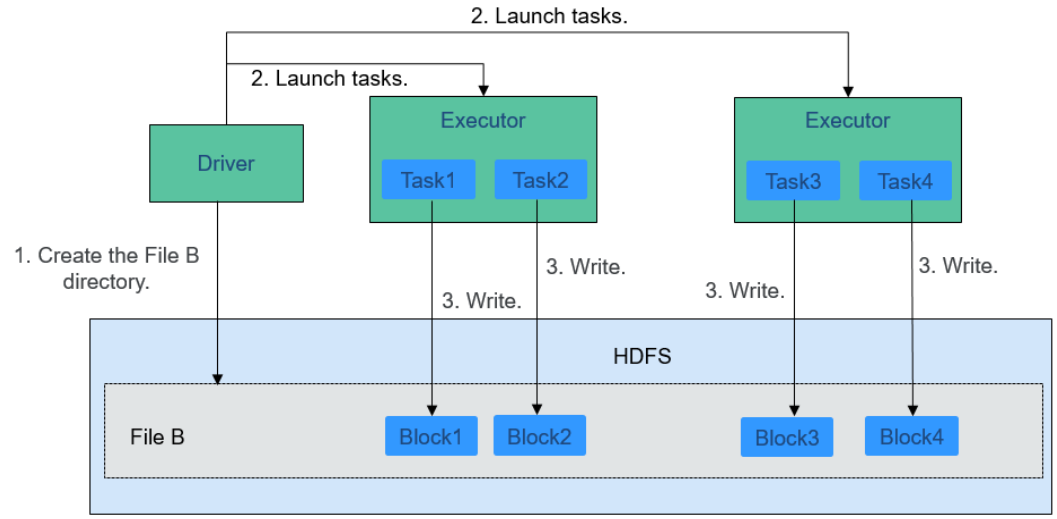


The file reading process is as follows:

1. Driver interconnects with HDFS to obtain the information of File A.
2. The HDFS returns the detailed block information about this file.
3. Driver sets a parallel degree based on the block data amount, and creates multiple tasks to read the blocks of this file.
4. Executor runs the tasks and reads the detailed blocks as part of the Resilient Distributed Dataset (RDD).

Figure 1-48 shows how data is written to a file.

Figure 1-48 File writing process



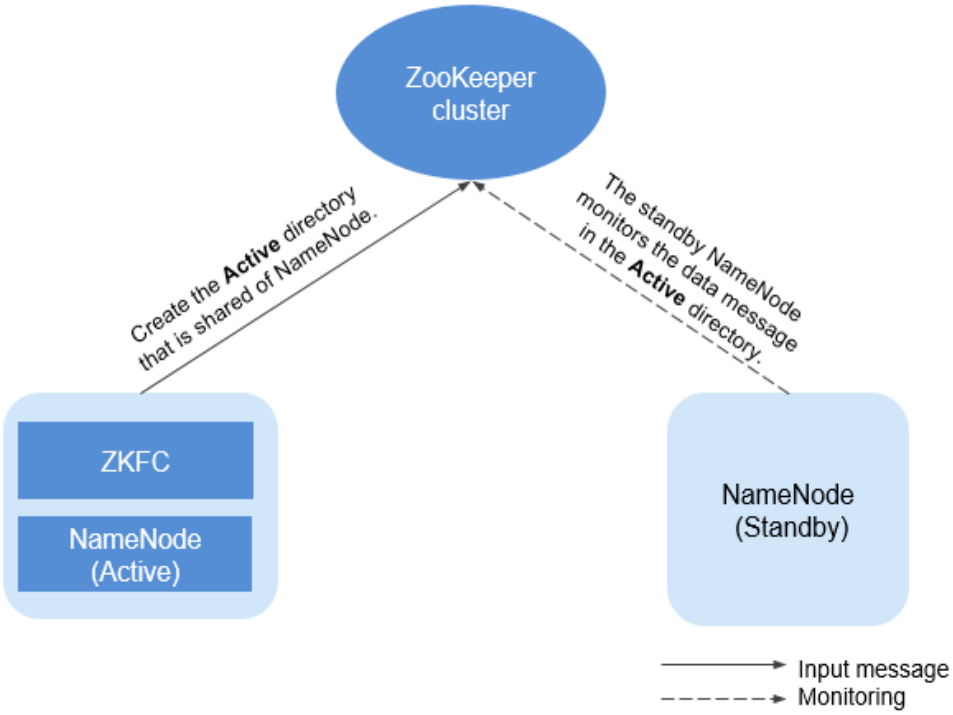
The file writing process is as follows:

1. Driver creates a directory where the file is to be written.
2. Based on the RDD distribution status, the number of tasks related to data writing is computed, and these tasks are sent to Executor.
3. Executor runs these tasks, and writes the computed RDD data to the directory created in 1.

Relationship Between HDFS and ZooKeeper

Figure 1-49 shows the relationship between ZooKeeper and HDFS.

Figure 1-49 Relationship between ZooKeeper and HDFS



As the client of a ZooKeeper cluster, ZKFailoverController (ZKFC) monitors the status of NameNode. ZKFC is deployed only in the node where NameNode resides, and in both the active and standby HDFS NameNodes.

1. The ZKFC connects to ZooKeeper and saves information such as host names to ZooKeeper under the znode directory **/hadoop-ha**. NameNode that creates the directory first is considered as the active node, and the other is the standby node. NameNodes read the NameNode information periodically through ZooKeeper.
2. When the process of the active node ends abnormally, the standby NameNode detects changes in the **/hadoop-ha** directory through ZooKeeper, and then takes over the service of the active NameNode.

1.4.8.4 HDFS Enhanced Open Source Features

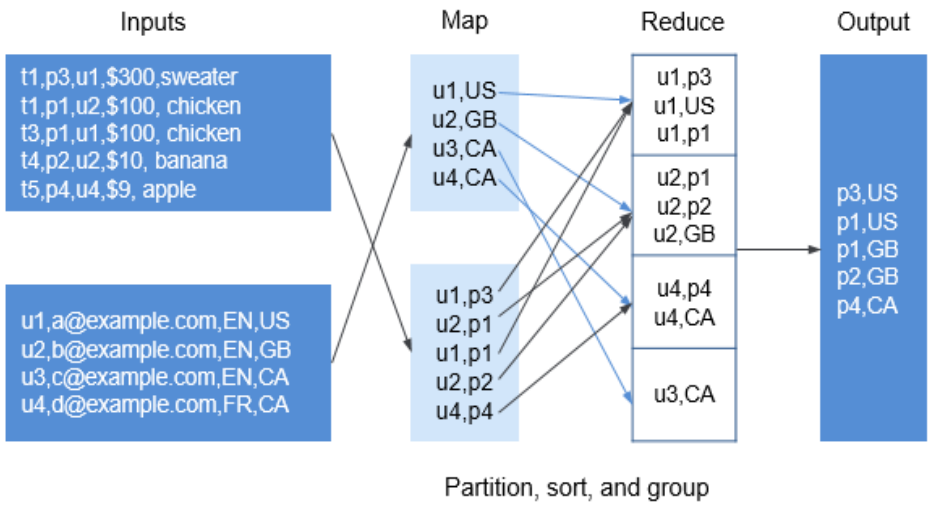
Enhanced Open Source Feature: File Block Colocation

In the offline data summary and statistics scenario, Join is a frequently used computing function, and is implemented in MapReduce as follows:

1. The Map task processes the records in the two table files into Join Key and Value, performs hash partitioning by Join Key, and sends the data to different Reduce tasks for processing.
2. Reduce tasks read data in the left table recursively in the nested loop mode and traverse each line of the right table. If join key values are identical, join results are output.

   The preceding method sharply reduces the performance of the join calculation. Because a large amount of network data transfer is required when the data stored in different nodes is sent from MAP to Reduce, as shown in **Figure 1-50**.

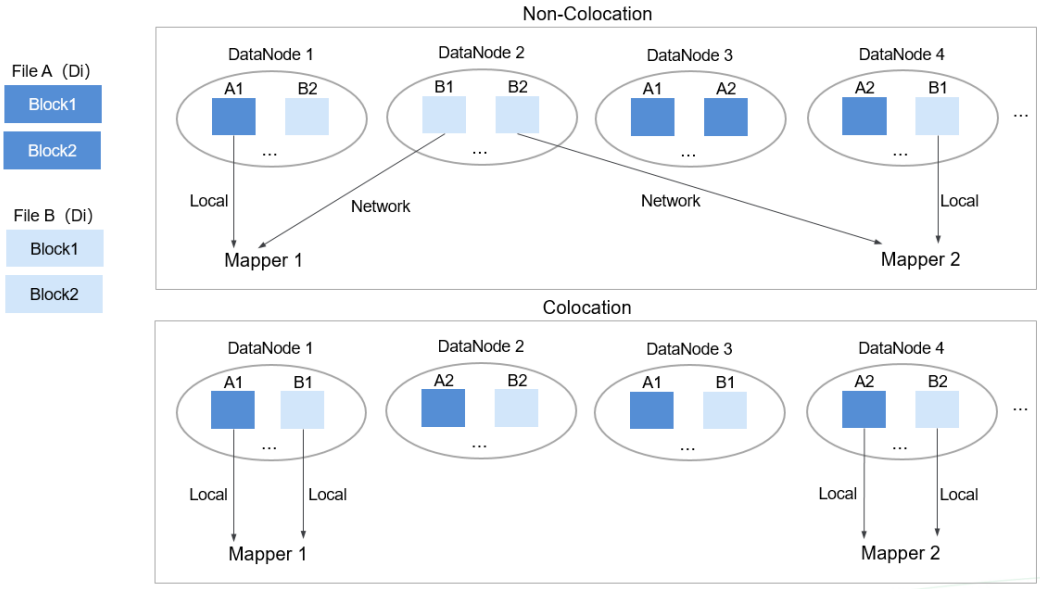
Figure 1-50 Data transmission in the non-colocation scenario



Data tables are stored in physical file system by HDFS block. Therefore, if two to-be-joined blocks are put into the same host accordingly after they are partitioned by join key, you can obtain the results directly from Map join in the local node without any data transfer in the Reduce process of the join calculation. This will greatly improve the performance.

With the identical distribution feature of HDFS data, a same distribution ID is allocated to files, FileA and FileB, on which association and summation calculations need to be performed. In this way, all the blocks are distributed together, and calculation can be performed without retrieving data across nodes, which greatly improves the MapReduce join performance.

Figure 1-51 Data block distribution in colocation and non-colocation scenarios

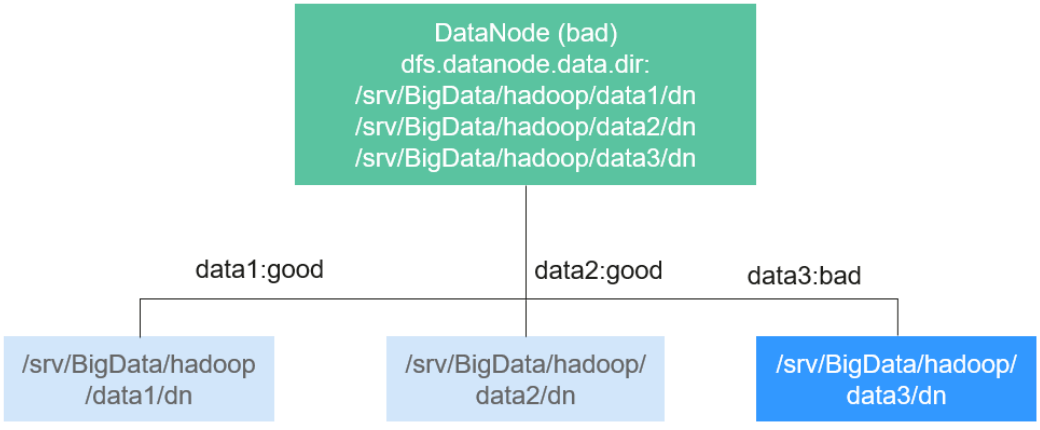


Enhanced Open Source Feature: Damaged Hard Disk Volume Configuration

In the open source version, if multiple data storage volumes are configured for a DataNode, the DataNode stops providing services by default if one of the volumes is damaged. If the configuration item **dfs.datanode.failed.volumes.tolerated** is set to specify the number of damaged volumes that are allowed, DataNode continues to provide services when the number of damaged volumes does not exceed the threshold.

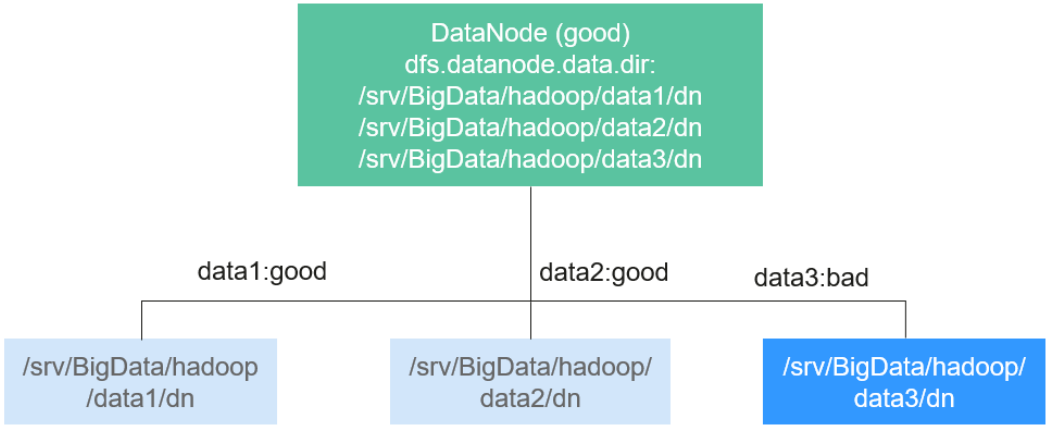
The value of **dfs.datanode.failed.volumes.tolerated** ranges from -1 to the number of disk volumes configured on the DataNode. The default value is **-1**, as shown in **Figure 1-52**.

Figure 1-52 Item being set to 0



For example, three data storage volumes are mounted to a DataNode, and **dfs.datanode.failed.volumes.tolerated** is set to 1. In this case, if one data storage volume of the DataNode is unavailable, this DataNode can still provide services, as shown in **Figure 1-53**.

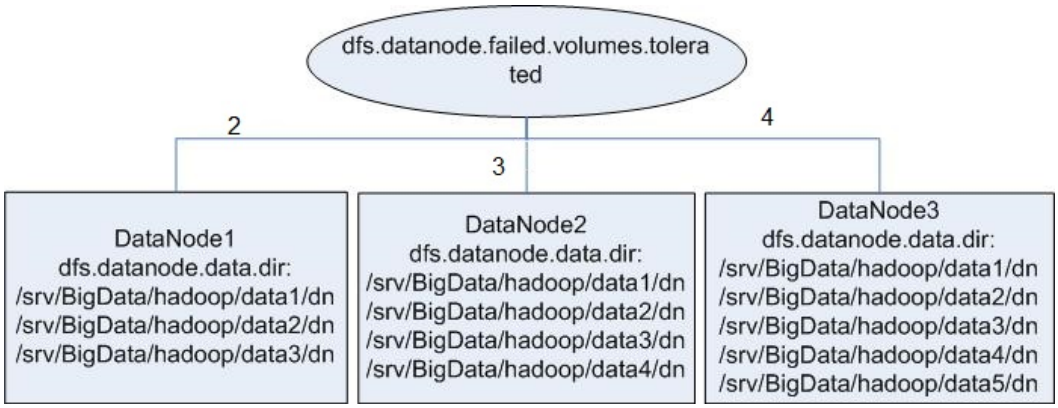
Figure 1-53 Item being set to 1



This native configuration item has some defects. When the number of data storage volumes in each DataNode is inconsistent, you need to configure each DataNode independently instead of generating the unified configuration file for all nodes.

Assume that there are three DataNodes in a cluster. The first node has three data directories, the second node has four, and the third node has five. If you want to ensure that DataNode services are available when only one data directory is available, you need to perform the configuration as shown in **Figure 1-54**.

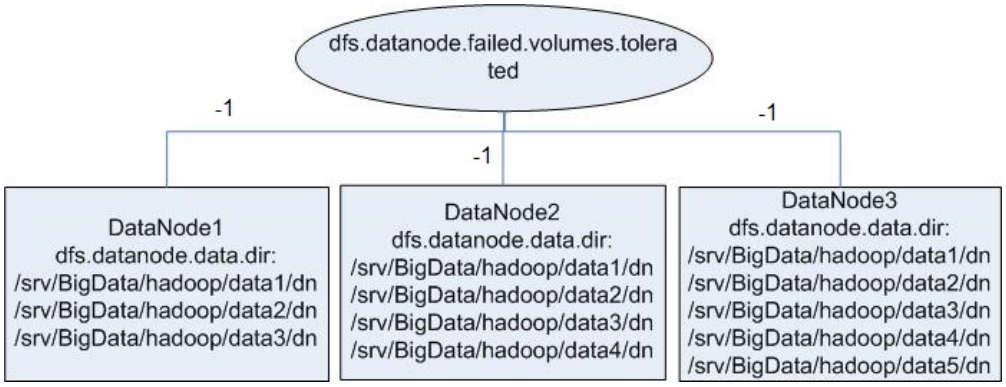
Figure 1-54 Attribute configuration before being enhanced



In self-developed enhanced HDFS, this configuration item is enhanced, with a value **-1** added. When this configuration item is set to **-1**, all DataNodes can provide services as long as one data storage volume in all DataNodes is available.

To resolve the problem in the preceding example, set this configuration to **-1**, as shown in **Figure 1-55**.

Figure 1-55 Attribute configuration after being enhanced



Enhanced Open Source Feature: HDFS Startup Acceleration

In HDFS, when NameNodes start, the metadata file FsImage needs to be loaded. Then, DataNodes will report the data block information after the DataNodes startup. When the data block information reported by DataNodes reaches the preset percentage, NameNodes exits safe mode to complete the startup process. If the number of files stored on the HDFS reaches the million or billion level, the two processes are time-consuming and will lead to a long startup time of the NameNode. Therefore, this version optimizes the process of loading metadata file FsImage.

In the open source HDFS, FsImage stores all types of metadata information. Each type of metadata information (such as file metadata information and folder metadata information) is stored in a section block, respectively. These section blocks are loaded in serial mode during startup. If a large number of files and folders are stored on the HDFS, loading of the two sections is time-consuming, prolonging the HDFS startup time. HDFS NameNode divides each type of metadata by segments and stores the data in multiple sections when generating the FsImage files. When the NameNodes start, sections are loaded in parallel mode. This accelerates the HDFS startup.

Enhanced Open Source Feature: Label-based Block Placement Policies (HDFS Nodelabel)

You need to configure the nodes for storing HDFS file data blocks based on data features. You can configure a label expression to an HDFS directory or file and assign one or more labels to a DataNode so that file data blocks can be stored on specified DataNodes. If the label-based data block placement policy is used for selecting DataNodes to store the specified files, the DataNode range is specified based on the label expression. Then proper nodes are selected from the specified range.

- You can store the replicas of data blocks to the nodes with different labels accordingly. For example, store two replicas of the data block to the node labeled with L1, and store other replicas of the data block to the nodes labeled with L2.
- You can set the policy in case of block placement failure, for example, select a node from all nodes randomly.

Figure 1-56 gives an example:

- Data in **/HBase** is stored in A, B, and D.
- Data in **/Spark** is stored in A, B, D, E, and F.
- Data in **/user** is stored in C, D, and F.
- Data in **/user/shl** is stored in A, E, and F.

Figure 1-56 Example of label-based block placement policy

Enhanced Open Source Feature: HDFS Load Balance

The current read and write policies of HDFS are mainly for local optimization without considering the actual load of nodes or disks. Based on I/O loads of different nodes, the load balance of HDFS ensures that when read and write operations are performed on the HDFS client, the node with low I/O load is selected to perform such operations to balance I/O load and fully utilize the overall throughput of the cluster.

If HDFS Load Balance is enabled during file writing, the NameNode selects a DataNode (in the order of local node, local rack, and remote rack). If the I/O load of the selected node is heavy, the NameNode will choose another DataNode with lighter load.

If HDFS Load Balance is enabled during file reading, an HDFS client sends a request to the NameNode to provide the list of DataNodes that store the block to be read. The NameNode returns a list of DataNodes sorted by distance in the network topology. With the HDFS Load Balance feature, the DataNodes on the list

are also sorted by their I/O load. The DataNodes with heavy load are at the bottom of the list.

Enhanced Open Source Feature: HDFS Auto Data Movement

Hadoop has been used for batch processing of immense data in a long time. The existing HDFS model is used to fit the needs of batch processing applications very well because such applications focus more on throughput than delay.

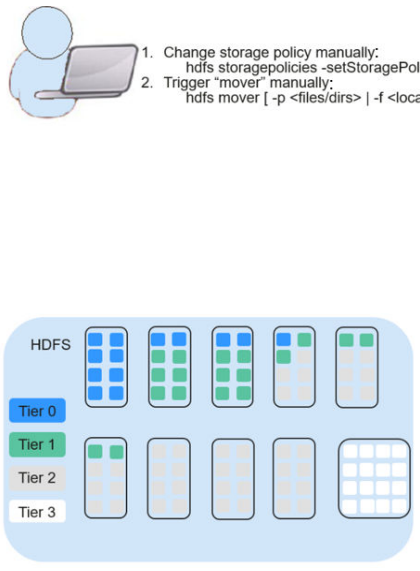
However, as Hadoop is increasingly used for upper-layer applications that demand frequent random I/O access such as Hive and HBase, low latency disks such as solid state disk (SSD) are favored in delay-sensitive scenarios. To cater to the trend, HDFS supports a variety of storage types. Users can choose a storage type according to their needs.

Storage policies vary depending on how frequently data is used. For example, if data that is frequently accessed in the HDFS is marked as **ALL_SSD** or **HOT**, the data that is accessed several times may be marked as **WARM**, and data that is rarely accessed (only once or twice access) can be marked as **COLD**. You can select different data storage policies based on the data access frequency.

However, low latency disks are far more expensive than spinning disks. Data typically sees heavy initial usage with decline in usage over a period of time. Therefore, it can be useful if data that is no longer used is moved out from expensive disks to cheaper ones storage media.

A typical example is storage of detail records. New detail records are imported into SSD because they are frequently queried by upper-layer applications. As access frequency to these detail records declines, they are moved to cheaper storage.

Before automatic data movement is achieved, you have to manually determine by service type whether data is frequently used, manually set a data storage policy, and manually trigger the HDFS Auto Data Movement Tool, as shown in the figure below.



1. Change storage policy manually:
   hdfs storagepolicies -setStoragePolicy –path <path> -policy <policy>
2. Trigger "mover" manually:
   hdfs mover [-p <files/dirs> | -f <local file name>]

Policy ID	PolicyName	Block Placement (n replacas)	Fallback storages for creation	Fallback storages for replication
15	Lazy_Persist	RAN_DISK:1 DISK:n-1	DISK	DISK
12	All_SSD	SSD:n	DISK	DISK
10	One_SSD	SSD:1,DISK:n-1	SSD,DISK	SSD,DISK
7	Hot(default)	DISK:n	<none>	ARCHIVE
5	Warm	DISK:1,ARCHIVE:n-1	ARCHIVE, DISK	ARCHIVE, DISK
2	Cold	ARCHIVE:n	<none>	<none>

If aged data can be automatically identified and moved to cheaper storage (such as disk/archive), you will see significant cost cuts and data management efficiency improvement.

The HDFS Auto Data Movement Tool is at the core of HDFS Auto Data Movement. It automatically sets a storage policy depending on how frequently data is used. Specifically, functions of the HDFS Auto Data Movement Tool can:

- Mark a data storage policy as **All_SSD**, **One_SSD**, **Hot**, **Warm**, **Cold**, or **FROZEN** according to age, access time, and manual data movement rules.
- Define rules for distinguishing cold and hot data based on the data age, access time, and manual migration rules.
- Define the action to be taken if age-based rules are met.

  MARK: the action for identifying whether data is frequently or rarely used based on the age rules and setting a data storage policy. **MOVE**: the action for invoking the HDFS Auto Data Movement Tool and moving data based on the age rules to identify whether data is frequently or rarely used after you have determined the corresponding storage policy.

  - **MARK**: identifies whether data is frequently or rarely used and sets the data storage policy.
  - **MOVE**: the action for invoking the HDFS Auto Data Movement Tool and moving data across tiers.
  - **SET_REPL**: the action for setting new replica quantity for a file.
  - **MOVE_TO_FOLDER**: the action for moving files to a target folder.
  - **DELETE**: the action for deleting a file or directory.
  - **SET_NODE_LABEL**: the action for setting node labels of a file.

With the HDFS Auto Data Movement feature, you only need to define age based on access time rules. HDFS Auto Data Movement Tool matches data according to age-based rules, sets storage policies, and moves data. In this way, data management efficiency and cluster resource efficiency are improved.

1.4.9 Hive

1.4.9.1 Hive Basic Principles

Hive is a data warehouse infrastructure built on Hadoop. It provides a series of tools that can be used to extract, transform, and load (ETL) data. Hive is a mechanism that can store, query, and analyze mass data stored on Hadoop. Hive defines simple SQL-like query language, which is known as HiveQL. It allows a user familiar with SQL to query data. Hive data computing depends on MapReduce, Spark, and Tez.

The new execution engine **Tez** is used to replace the original MapReduce, greatly improving performance. Tez can convert multiple dependent jobs into one job, so only once HDFS write is required and fewer transit nodes are needed, greatly improving the performance of DAG jobs.

Hive provides the following functions:

- Analyzes massive structured data and summarizes analysis results.

- Allows complex MapReduce jobs to be compiled in SQL languages.
- Supports flexible data storage formats, including JavaScript object notation (JSON), comma separated values (CSV), TextFile, RCFile, SequenceFile, and ORC (Optimized Row Columnar).

Hive system structure:

- User interface: Three user interfaces are available, that is, CLI, Client, and WUI. CLI is the most frequently-used user interface. A Hive transcript is started when CLI is started. Client refers to a Hive client, and a client user connects to the Hive Server. When entering the client mode, you need to specify the node where the Hive Server resides and start the Hive Server on this node. The web UI is used to access Hive through a browser. MRS can access Hive only in client mode.
- Metadata storage: Hive stores metadata into databases, for example, MySQL and Derby. Metadata in Hive includes a table name, table columns and partitions and their properties, table properties (indicating whether a table is an external table), and the directory where table data is stored.

Hive Framework

Hive is a single-instance service process that provides services by translating HQL into related MapReduce jobs or HDFS operations. **Figure 1-57** shows how Hive is connected to other components.

Figure 1-57 Hive framework

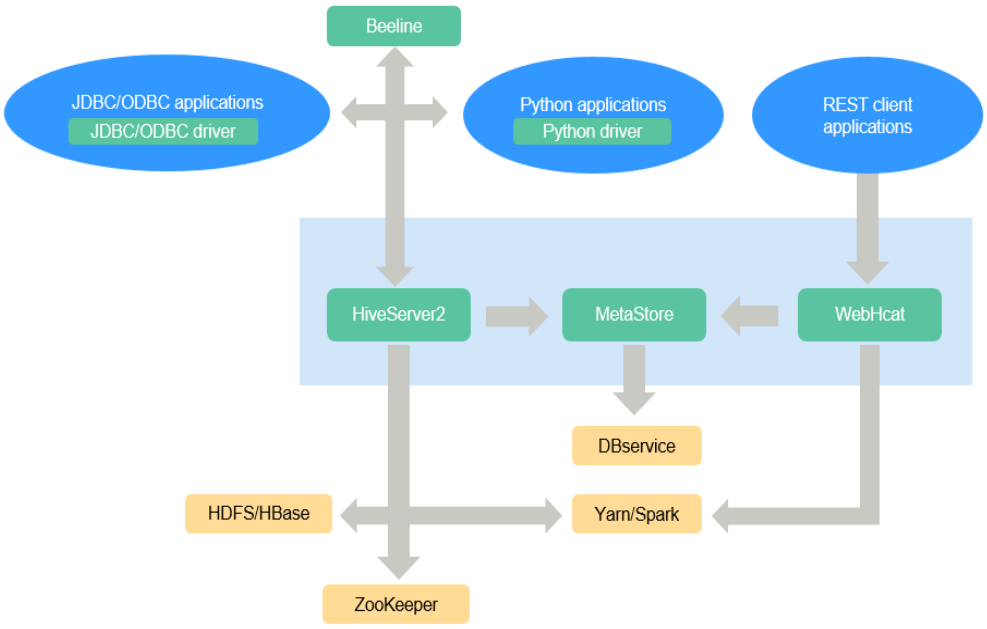
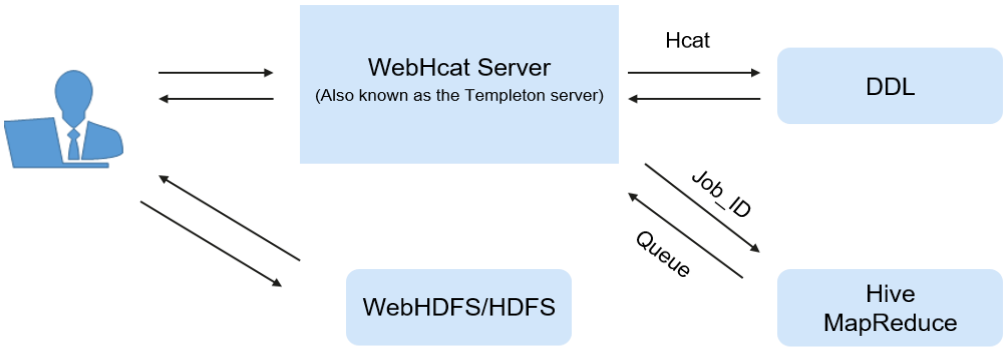


Table 1-9 Module description

Module	Description
HiveServer	Multiple HiveServers can be deployed in a cluster to share loads. HiveServer provides Hive database services externally, translates HQL statements into related YARN tasks or HDFS operations to complete data extraction, conversion, and analysis.
MetaStore	• Multiple MetaStores can be deployed in a cluster to share loads. MetaStore provides Hive metadata services as well as reads, writes, maintains, and modifies the structure and properties of Hive tables. • MetaStore provides Thrift APIs for HiveServer, Spark, WebHCat, and other MetaStore clients to access and operate metadata.
WebHCat	Multiple WebHCats can be deployed in a cluster to share loads. WebHCat provides REST APIs and runs the Hive commands through the REST APIs to submit MapReduce jobs.
Hive client	Hive client includes the human-machine command-line interface (CLI) Beeline, JDBC drive for JDBC applications, Python driver for Python applications, and HCatalog JAR files for MapReduce.
ZooKeeper cluster	As a temporary node, ZooKeeper records the IP address list of each HiveServer instance. The client driver connects to ZooKeeper to obtain the list and selects corresponding HiveServer instances based on the routing mechanism.
HDFS/HBase cluster	The HDFS cluster stores the Hive table data.
MapReduce/ YARN cluster	Provides distributed computing services. Most Hive data operations rely on MapReduce. The main function of HiveServer is to translate HQL statements into MapReduce jobs to process massive data.

HCatalog is built on Hive Metastore and incorporates the DDL capability of Hive. HCatalog is also a Hadoop-based table and storage management layer that enables convenient data read/write on tables of HDFS by using different data processing tools such as MapReduce. Besides, HCatalog also provides read/write APIs for these tools and uses a Hive CLI to publish commands for defining data and querying metadata. After encapsulating these commands, WebHCat Server can provide RESTful APIs, as shown in **Figure 1-58**.

Figure 1-58 WebHCat logical architecture



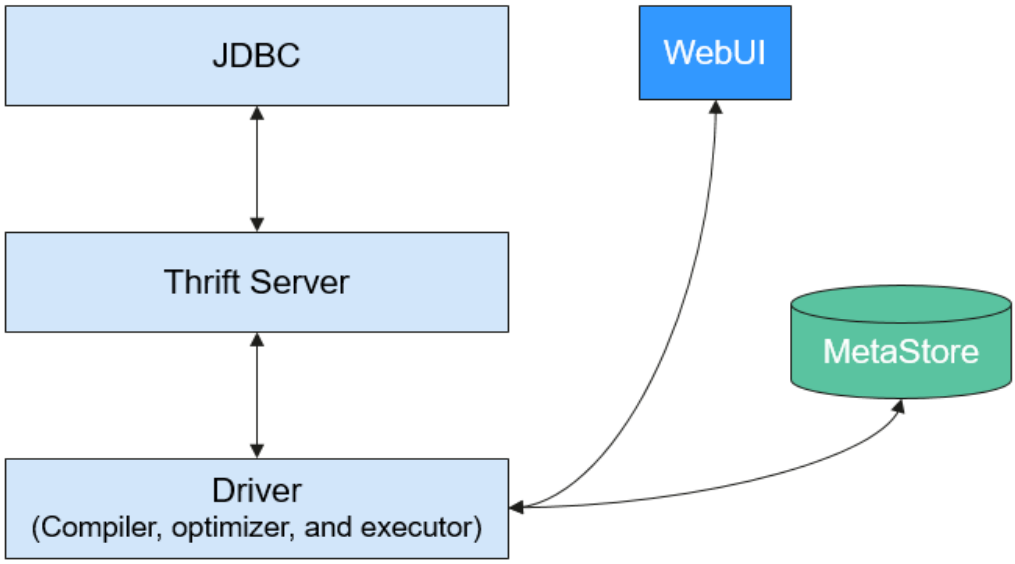
Principles

Hive functions as a data warehouse based on HDFS and MapReduce architecture and translates HQL statements into MapReduce jobs or HDFS operations. For details about Hive and HQL, see **HiveQL Language Manual**.

Figure 1-59 shows the Hive structure.

- **Metastore**: reads, writes, and updates metadata such as tables, columns, and partitions. Its lower layer is relational databases.
- **Driver**: manages the lifecycle of HiveQL execution and participates in the entire Hive job execution.
- **Compiler**: translates HQL statements into a series of interdependent Map or Reduce jobs.
- **Optimizer**: is classified into logical optimizer and physical optimizer to optimize HQL execution plans and MapReduce jobs, respectively.
- **Executor**: runs Map or Reduce jobs based on job dependencies.
- **ThriftServer**: functions as the servers of JDBC, provides Thrift APIs, and integrates with Hive and other applications.
- **Clients**: include the WebUI and JDBC APIs and provides APIs for user access.

Figure 1-59 Hive framework



1.4.9.2 Hive CBO Principles

Hive CBO Principles

CBO is short for Cost-Based Optimization.

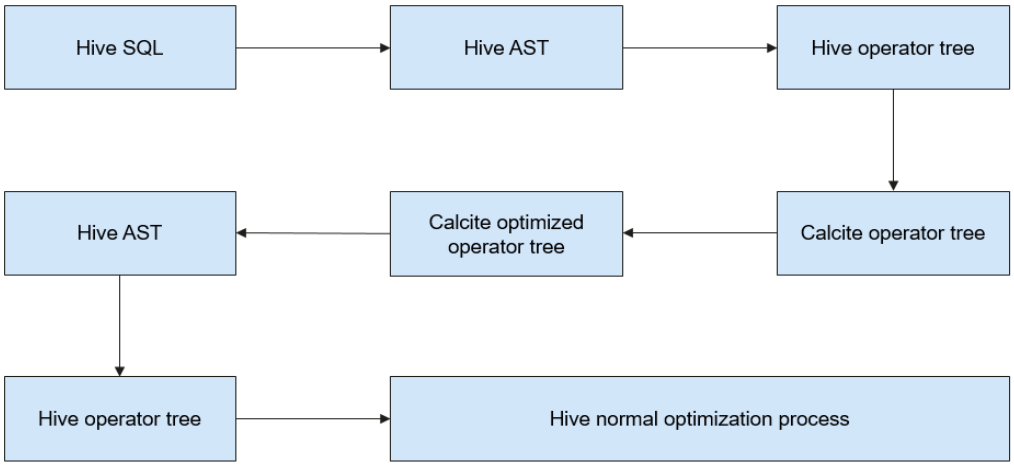
It will optimize the following:

During compilation, the CBO calculates the most efficient join sequence based on tables and query conditions involved in query statements to reduce time and resources required for query.

In Hive, the CBO is implemented as follows:

Hive uses open-source component Apache Calcite to implement the CBO. SQL statements are first converted into Hive Abstract Syntax Trees (ASTs) and then into RelNodes that can be identified by Calcite. After Calcite adjusts the join sequence in RelNodes, RelNodes are converted into ASTs by Hive to continue the logical and physical optimization. **Figure 1-60** shows the working flow.

Figure 1-60 CBO Implementation process



Calcite adjusts the join sequence as follows:

1. A table is selected as the first table from the tables to be joined.
2. The second and third tables are selected based on the cost. In this way, multiple different execution plans are obtained.
3. A plan with the minimum costs is calculated and serves as the final sequence.

The cost calculation method is as follows:

In the current version, costs are measured based on the number of data entries after joining. Fewer data entries mean less cost. The number of joined data entries depends on the selection rate of joined tables. The number of data entries in a table is obtained based on the table-level statistics.

The number of data entries in a table after filtering is estimated based on the column-level statistics, including the maximum values (max), minimum values (min), and Number of Distinct Values (NDV).

For example, there is a table **table_a** whose total number of data records is 1,000,000 and NDV is 50. The query conditions are as follows:

```
Select * from table_a where colum_a='value1';
```

The estimated number of queried data entries is: 1,000,000 x 1/50 = 20,000. The selection rate is 2%.

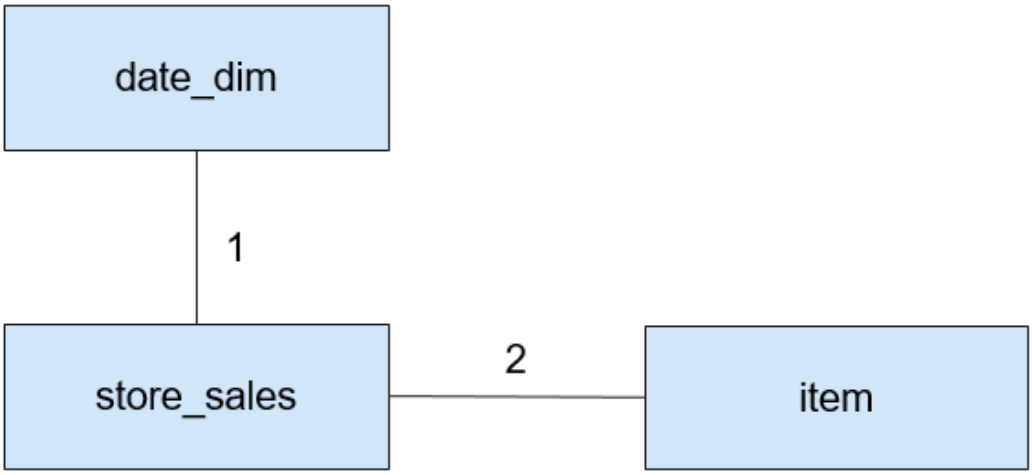
The following takes the TPC-DS Q3 as an example to describe how the CBO adjusts the join sequence:

```
select
    dt.d_year,
    item.i_brand_id brand_id,
    item.i_brand brand,
    sum(ss_ext_sales_price) sum_agg
from
    date_dim dt,
    store_sales,
    item
where
    dt.d_date_sk = store_sales.ss_sold_date_sk
    and store_sales.ss_item_sk = item.i_item_sk
```

```
   and item.i_manufact_id = 436
   and dt.d_moy = 12
group by dt.d_year , item.i_brand , item.i_brand_id
order by dt.d_year , sum_agg desc , brand_id
limit 10;
```

Statement explanation: This statement indicates that inner join is performed for three tables: table **store_sales** is a fact table with about 2,900,000,000 data entries, table **date_dim** is a dimension table with about 73,000 data entries, and table **item** is a dimension table with about 18,000 data entries. Each table has filtering conditions. **Figure 1-61** shows the join relationship.

Figure 1-61 Join relationship



The CBO must first select the tables that bring the best filtering effect for joining.

By analyzing min, max, NDV, and the number of data entries, the CBO estimates the selection rates of different dimension tables, as shown in **Table 1-10**.

Table 1-10 Data filtering

Table	Number of Original Data Entries	Number of Data Entries After Filtering	Selection Rate
date_dim	73,000	6,200	8.5%
item	18,000	19	0.1%

The selection rate can be estimated as follows: Selection rate = Number of data entries after filtering/Number of original data entries

As shown in the preceding table, the **item** table has a better filtering effect. Therefore, the CBO joins the **item** table first before joining the **date_dim** table.

Figure 1-62 shows the join process when the CBO is disabled.

Figure 1-62 Join process when the CBO is disabled

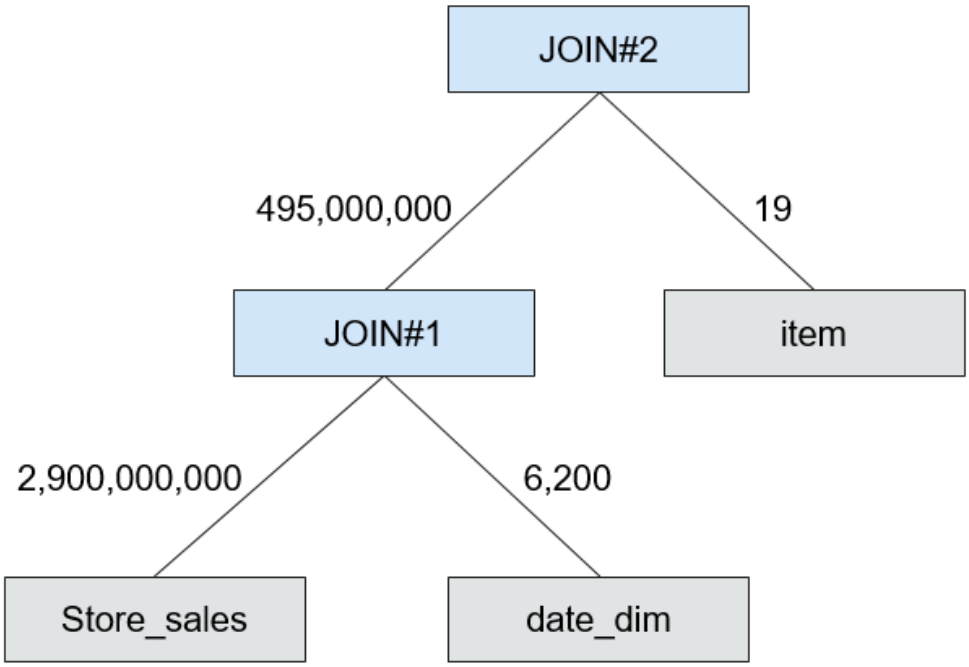
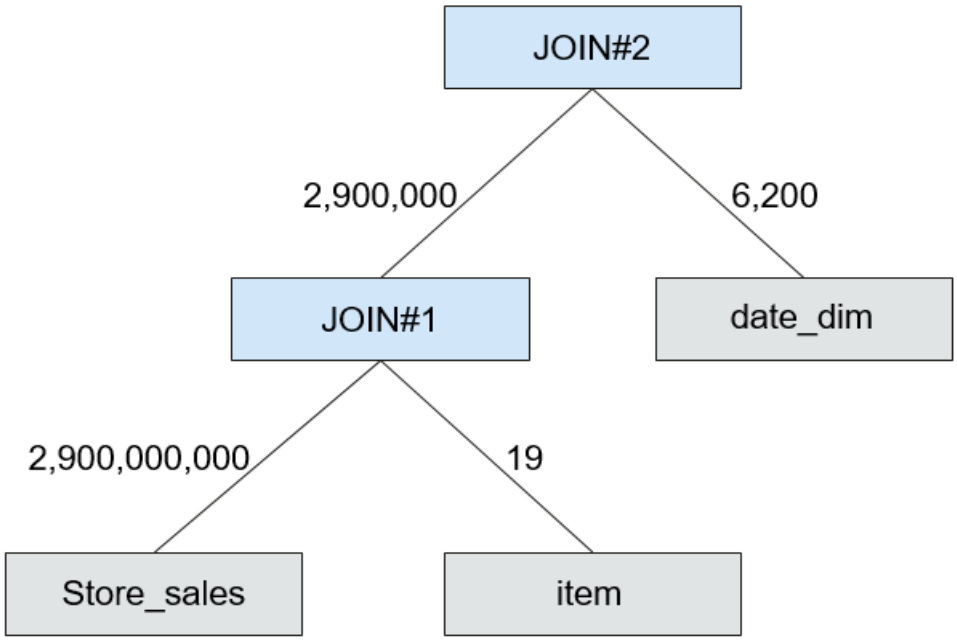


Figure 1-63 shows the join process when the CBO is enabled.

Figure 1-63 Join process when the CBO is enabled



After the CBO is enabled, the number of intermediate data entries is reduced from 495,000,000 to 2,900,000 and thus the execution time can be remarkably reduced.

1.4.9.3 Relationship Between Hive and Other Components

Relationship Between Hive and HDFS

Hive is a sub-project of Apache Hadoop, which uses HDFS as the file storage system. It parses and processes structured data with highly reliable underlying storage supported by HDFS. All data files in the Hive database are stored in HDFS, and all data operations on Hive are also performed using HDFS APIs.

Relationship Between Hive and MapReduce

Hive data computing depends on MapReduce. MapReduce is also a sub-project of Apache Hadoop and is a parallel computing framework based on HDFS. During data analysis, Hive parses HQL statements submitted by users into MapReduce tasks and submits the tasks for MapReduce to execute.

Relationship Between Hive and Tez

Tez, an open-source project of Apache, is a distributed computing framework that supports directed acyclic graphs (DAGs). When Hive uses the Tez engine to analyze data, it parses HQL statements submitted by users into Tez tasks and submits the tasks to Tez for execution.

Relationship Between Hive and DBService

MetaStore (metadata service) of Hive processes the structure and attribute information of Hive metadata, such as Hive databases, tables, and partitions. The information needs to be stored in a relational database and is managed and processed by MetaStore. In the product, the metadata of Hive is stored and maintained by the DBService component, and the metadata service is provided by the Metadata component.

1.4.9.4 Enhanced Open Source Feature

Enhanced Open Source Feature: HDFS Colocation

HDFS Colocation is the data location control function provided by HDFS. The HDFS Colocation API stores associated data or data on which associated operations are performed on the same storage node.

Hive supports HDFS Colocation. When Hive tables are created, after the locator information is set for table files, the data files of related tables are stored on the same storage node. This ensures convenient and efficient data computing among associated tables.

Enhanced Open Source Feature: Column Encryption

Hive supports encryption of one or more columns. The columns to be encrypted and the encryption algorithm can be specified when a Hive table is created. When data is inserted into the table using the INSERT statement, the related columns are encrypted. The Hive column encryption does not support views and the Hive over HBase scenario.

The Hive column encryption mechanism supports two encryption algorithms that can be selected to meet site requirements during table creation:

- AES (the encryption class is **org.apache.hadoop.hive.serde2.AESRewriter**)
- SMS4 (the encryption class is **org.apache.hadoop.hive.serde2.SMS4Rewriter**)

Enhanced Open Source Feature: HBase Deletion

Due to the limitations of underlying storage systems, Hive does not support the ability to delete a single piece of table data. In Hive on HBase, Hive in the MRS solution supports the ability to delete a single piece of HBase table data. Using a specific syntax, Hive can delete one or more pieces of data from an HBase table.

Enhanced Open Source Feature: Row Delimiter

In most cases, a carriage return character is used as the row delimiter in Hive tables stored in text files, that is, the carriage return character is used as the terminator of a row during queries.

However, some data files are delimited by special characters, and not a carriage return character.

MRS Hive allows you to specify different characters or character combinations as row delimiters for Hive data in text files.

Enhanced Open Source Feature: HTTPS/HTTP-based REST API Switchover

WebHCat provides external REST APIs for Hive. By default, the open source community version uses the HTTP protocol.

MRS Hive supports the HTTPS protocol that is more secure, and enables switchover between the HTTP protocol and the HTTPS protocol.

Enhanced Open Source Feature: Transform Function

The Transform function is not allowed by Hive of the open source version. MRS Hive supports the configuration of the Transform function. The function is disabled by default, which is the same as that of the open source community version.

Users can modify configurations of the Transform function to enable the function. However, security risks exist when the Transform function is enabled.

Enhanced Open Source Feature: Temporary Function Creation Without ADMIN Permission

You must have **ADMIN** permission when creating temporary functions on Hive of the open source community version. MRS Hive supports the configuration of the function for creating temporary functions with **ADMIN** permission. The function is disabled by default, which is the same as that of the open-source community version.

You can modify configurations of this function. After the function is enabled, you can create temporary functions without **ADMIN** permission.

Enhanced Open Source Feature: Database Authorization

In the Hive open source community version, only the database owner can create tables in the database. You can be granted with the **CREATE** and **SELECT** permissions on tables by MRS Hive in a database. After you are granted with the permission to query data in the database, the system automatically associates the query permission on all tables in the database.

Enhanced Open Source Feature: Column Authorization

The Hive open source community version supports only table-level permission control. MRS Hive supports column-level permission control. You can be granted with column-level permissions, such as **SELECT**, **INSERT**, and **UPDATE**.

1.4.10 Hue

1.4.10.1 Hue Basic Principles

Hue is a group of web applications that interact with MRS big data components. It helps you browse HDFS, perform Hive query, and start MapReduce jobs. Hue bears applications that interact with all MRS big data components.

Hue provides the file browser and query editor functions:

- File browser allows you to directly browse and operate different HDFS directories on the GUI.
- Query editor can write simple SQL statements to query data stored on Hadoop, for example, HDFS, HBase, and Hive. With the query editor, you can easily create, manage, and execute SQL statements and download the execution results as an Excel file.

On the WebUI provided by Hue, you can perform the following operations on the components:

- HDFS:
  - View, create, manage, rename, move, and delete files or directories.
  - File upload and download
  - Search for files, directories, file owners, and user groups; change the owners and permissions of the files and directories.
  - Manually configure HDFS directory storage policies and dynamic storage policies.
- Hive:
  - Edit and execute SQL/HQL statements. Save, copy, and edit the SQL/HQL template. Explain SQL/HQL statements. Save the SQL/HQL statement and query it.
  - Database presentation and data table presentation
  - Supporting different types of Hadoop storage
  - Use MetaStore to add, delete, modify, and query databases, tables, and views.

NOTE

If Internet Explorer is used to access the Hue page to execute HiveSQL statements, the execution fails, because the browser has functional problems. You are advised to use a compatible browser, for example, Google Chrome.

- Impala:
  - Edit and execute SQL/HQL statements. Save, copy, and edit the SQL/HQL template. Explain SQL/HQL statements. Save the SQL/HQL statement and query it.
  - Database presentation and data table presentation
  - Supporting different types of Hadoop storage
  - Use MetaStore to add, delete, modify, and query databases, tables, and views.

    NOTE

    If Internet Explorer is used to access the Hue page to execute HiveSQL statements, the execution fails, because the browser has functional problems. You are advised to use a compatible browser, for example, Google Chrome.
- MapReduce: Check MapReduce tasks that are being executed or have been finished in the clusters, including their status, start and end time, and run logs.
- Oozie: Hue provides the Oozie job manager function, in this case, you can use Oozie in GUI mode.
- ZooKeeper: Hue provides the ZooKeeper browser function for you to use ZooKeeper in GUI mode.

For details about Hue, visit **https://gethue.com/**.

Architecture

Hue, adopting the MTV (Model-Template-View) design, is a web application program running on Django Python. (Django Python is a web application framework that uses open source codes.)

Hue consists of Supervisor Process and WebServer. Supervisor Process is the core Hue process that manages application processes. Supervisor Process and WebServer interact with applications on WebServer through Thrift/REST APIs, as shown in **Figure 1-64**.

Figure 1-64 Hue architecture

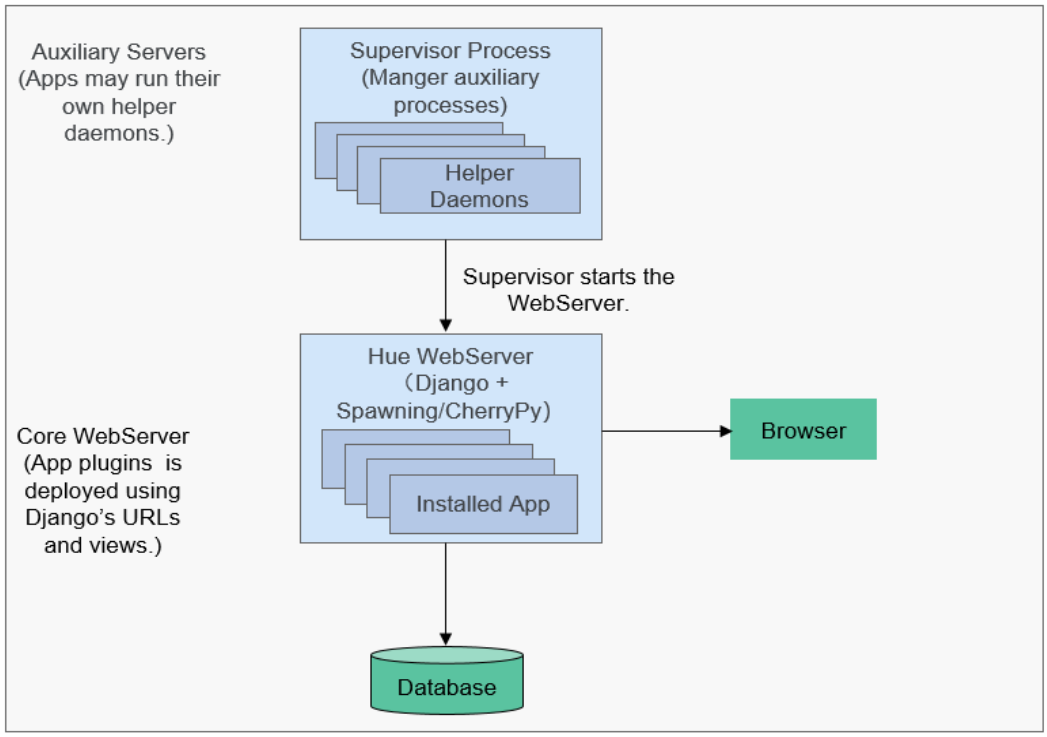


Table 1-11 describes the components shown in **Figure 1-64**.

Table 1-11 Architecture description

Connection Name	Description
Supervisor Process	Manages processes of WebServer applications, such as starting, stopping, and monitoring the processes.
Hue WebServer	Provides the following functions through the Django Python web framework: • Deploys applications. • Provides the GUI. • Connects to databases to store persistent data of applications.

1.4.10.2 Relationship Between Hue and Other Components

Relationship Between Hue and Hadoop Clusters

Figure 1-65 shows how Hue interacts with Hadoop clusters.

Figure 1-65 Hue and Hadoop clusters

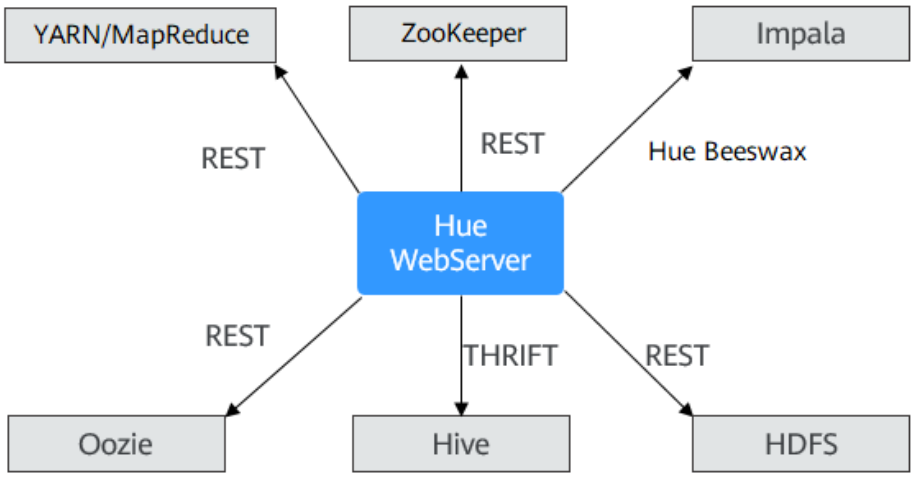


Table 1-12 Relationship Between Hue and Other Components

Connection Name	Description
HDFS	HDFS provides REST APIs to interact with Hue to query and operate HDFS files. Hue packages a user request into interface data, sends the request to HDFS through REST APIs, and displays execution results on the web UI.
Hive	Hive provides Thrift interfaces to interact with Hue, execute Hive SQL statements, and query table metadata. If you edit HQL statements on the Hue web UI, then, Hue submits the HQL statements to the Hive server through the Thrift APIs and displays execution results on the web UI.
YARN/ MapReduce	MapReduce provides REST APIs to interact with Hue and query YARN job information. If you go to the Hue web UI, enter the filter parameters, the UI sends the parameters to the background, and Hue invokes the REST APIs provided by MapReduce (MR1/MR2-YARN) to obtain information such as the status of the task running, the start/end time, the run log, and more.
Oozie	Oozie provides REST APIs to interact with Hue, create workflows, coordinators, and bundles, and manage and monitor tasks. A graphical workflow, coordinator, and bundle editor are provided on the Hue web UI. Hue invokes the REST APIs of Oozie to create, modify, delete, submit, and monitor workflows, coordinators, and bundles.

Connection Name	Description
ZooKeeper	ZooKeeper provides REST APIs to interact with Hue and query ZooKeeper node information. ZooKeeper node information is displayed in the Hue web UI. Hue invokes the REST APIs of ZooKeeper to obtain the node information.
Impala	Impala provides Hue Beeswax APIs to interact with Hue, execute Hive SQL statements, and query table metadata. If you edit HQL statements on the Hue web UI, then, Hue submits the HQL statements to the Hive server through the Hue Beeswax APIs and displays execution results on the web UI.

1.4.10.3 Hue Enhanced Open Source Features

Hue Enhanced Open Source Features

- Storage policy: The number of HDFS file copies varies depending on the storage media. This feature allows you to manually set an HDFS directory storage policy or can automatically adjust the file storage policy, modify the number of file copies, move the file directory, and delete files based on the latest access time and modification time of HDFS files to fully utilize storage capacity and improve storage performance.
- MR engine: You can use the MapReduce engine to execute Hive SQL statements.
- Reliability enhancement: Hue is deployed in active/standby mode. When interconnecting with HDFS, Oozie, Hive, and YARN, Hue can work in failover or load balancing mode.

1.4.11 Impala

Impala provides fast, interactive SQL queries directly on your Apache Hadoop data stored in HDFS, HBase, or the Object Storage Service (OBS). In addition to using the same unified storage platform, Impala also uses the same metadata, SQL syntax (Hive SQL), ODBC driver, and user interface (Impala query UI in Hue) as Apache Hive. This provides a familiar and unified platform for real-time or batch-oriented queries. Impala is an addition to tools available for querying big data. Impala does not replace the batch processing frameworks built on MapReduce such as Hive. Hive and other frameworks built on MapReduce are best suited for long running batch jobs.

Impala provides the following features:

- Most common SQL-92 features of Hive Query Language (HiveQL) including SELECT, JOIN, and aggregate functions
- HDFS, HBase, and OBS storage, including:
  - HDFS file formats: delimited text files, Parquet, Avro, SequenceFile, and RCFile

  - Compression codecs: Snappy, GZIP, Deflate, BZIP
- Common data access interfaces including:
  - JDBC driver
  - ODBC driver
  - Hue Beeswax and the Impala query UI
- **impala-shell** command line interface
- Kerberos authentication

Impala applies to offline analysis (such as log and cluster status analysis) of real-time data queries, large-scale data mining (such as user behavior analysis, interest region analysis, and region display), and other scenarios.

For details about Impala, visit **https://impala.apache.org/impala-docs.html**.

1.4.12 Kafka

1.4.12.1 Kafka Basic Principles

Kafka is an open source, distributed, partitioned, and replicated commit log service. Kafka is publish-subscribe messaging, rethought as a distributed commit log. It provides features similar to Java Message Service (JMS) but another design. It features message endurance, high throughput, distributed methods, multi-client support, and real time. It applies to both online and offline message consumption, such as regular message collection, website activeness tracking, aggregation of statistical system operation data (monitoring data), and log collection. These scenarios engage large amounts of data collection for Internet services.

Kafka Structure

Producers publish data to topics, and consumers subscribe to the topics and consume messages. A broker is a server in a Kafka cluster. For each topic, the Kafka cluster maintains partitions for scalability, parallelism, and fault tolerance. Each partition is an ordered, immutable sequence of messages that is continually appended to - a commit log. Each message in a partition is assigned a sequential ID, which is called offset.

Figure 1-66 Kafka architecture

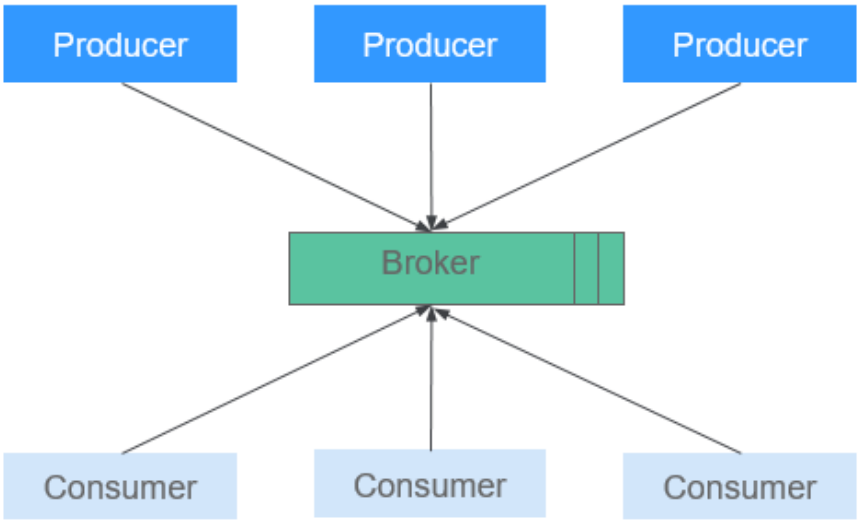
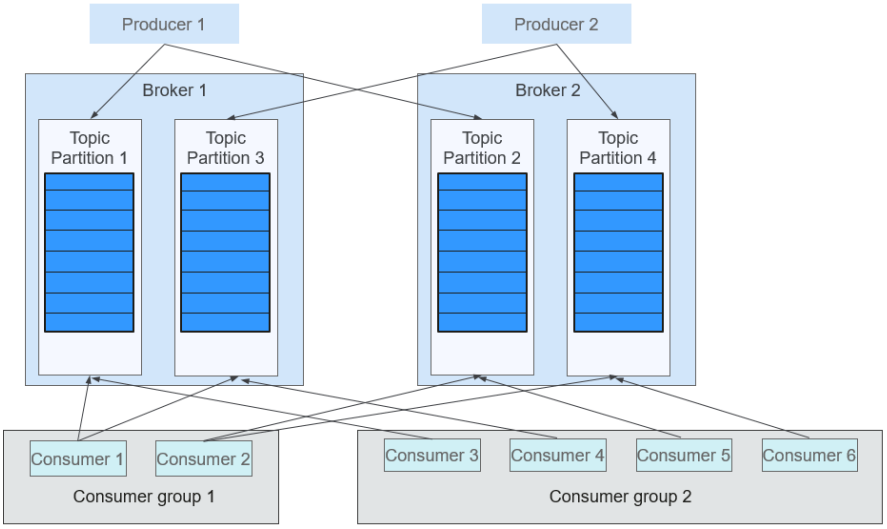


Table 1-13 Kafka architecture description

Name	Description
Broker	A broker is a server in a Kafka cluster.
Topic	A topic is a category or feed name to which messages are published. A topic can be divided into multiple partitions, which can act as a parallel unit.
Partition	A partition is an ordered, immutable sequence of messages that is continually appended to - a commit log. The messages in the partitions are each assigned a sequential ID number called the offset that uniquely identifies each message within the partition.
Producer	Producers publish messages to a Kafka topic.
Consumer	Consumers subscribe to topics and process the feed of published messages.

Figure 1-67 shows the relationships between modules.

Figure 1-67 Relationships between Kafka modules



Consumers label themselves with a consumer group name, and each message published to a topic is delivered to one consumer instance within each subscribing consumer group. If all the consumer instances belong to the same consumer group, loads are evenly distributed among the consumers. As shown in the preceding figure, Consumer1 and Consumer2 work in load-sharing mode; Consumer3, Consumer4, Consumer5, and Consumer6 work in load-sharing mode. If all the consumer instances belong to different consumer groups, messages are broadcast to all consumers. As shown in the preceding figure, the messages in Topic 1 are broadcast to all consumers in Consumer Group1 and Consumer Group2.

For details about Kafka architecture and principles, see **https://kafka.apache.org/24/documentation.html**.

Principle

- **Message Reliability**

  When a Kafka broker receives a message, it stores the message on a disk persistently. Each partition of a topic has multiple replicas stored on different broker nodes. If one node is faulty, the replicas on other nodes can be used.

- **High Throughput**

  Kafka provides high throughput in the following ways:

  - Messages are written into disks instead of being cached in the memory, fully utilizing the sequential read and write performance of disks.
  - The use of zero-copy eliminates I/O operations.
  - Data is sent in batches, improving network utilization.
  - Each topic is divided in to multiple partitions, which increases concurrent processing. Concurrent read and write operations can be performed between multiple producers and consumers. Producers send messages to specified partitions based on the algorithm used.

- **Message Subscribe-Notify Mechanism**

Consumers subscribe to interested topics and consume data in pull mode. Consumers can choose the consumption mode, such as batch consumption, repeated consumption, and consumption from the end, and control the message pulling speed based on actual situation. Consumers need to maintain the consumption records by themselves.

- **Scalability**

  When broker nodes are added to expand the Kafka cluster capacity, the newly added brokers register with ZooKeeper. After the registration is successful, procedures and consumers can sense the change in a timely manner and make related adjustment.

Open Source Features

- Reliability

  Message processing methods such as **At-Least Once**, **At-Most Once**, and **Exactly Once** are provided. The message processing status is maintained by consumers. Kafka needs to work with the application layer to implement **Exactly Once**.

- High throughput

  High throughput is provided for message publishing and subscription.

- Persistence

  Messages are stored on disks and can be used for batch consumption and real-time application programs. Data persistence and replication prevent data loss.

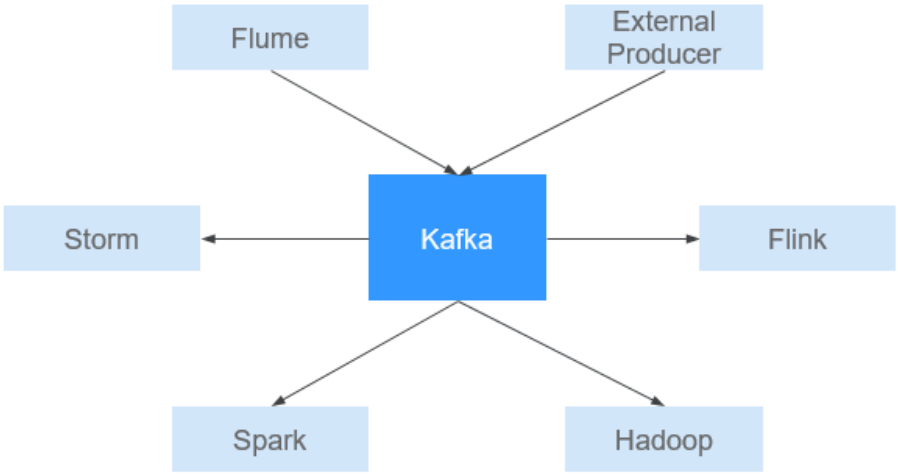
- Distribution

  A distributed system is easy to be expanded externally. All producers, brokers, and consumers support the deployment of multiple distributed clusters. Systems can be scaled without stopping the running of software or shutting down the machines.

1.4.12.2 Relationship Between Kafka and Other Components

As a message publishing and subscription system, Kafka provides high-speed data transmission methods for data transmission between different subsystems of the FusionInsight platform. It can receive external messages in a real-time manner and provides the messages to the online and offline services for processing. The following figure shows the relationship between Kafka and other components.

Figure 1-68 Relationship with Other Components



1.4.12.3 Kafka Enhanced Open Source Features

Kafka Enhanced Open Source Features

- Monitors the following topic-level metrics:
  - Topic Input Traffic
  - Topic Output Traffic
  - Topic Rejected Traffic
  - Number of Failed Fetch Requests Per Second
  - Number of Failed Produce Requests Per Second
  - Number of Topic Input Messages Per Second
  - Number of Fetch Requests Per Second
  - Number of Produce Requests Per Second
- Queries the mapping between broker IDs and node IP addresses. On Linux clients, **kafka-broker-info.sh** can be used to query the mapping between broker IDs and node IP addresses.

1.4.13 KafkaManager

KafkaManager is a tool for managing Apache Kafka and provides GUI-based metric monitoring and management of Kafka clusters.

KafkaManager supports the following operations:

- Manage multiple Kafka clusters.
- Easy inspection of cluster states (topics, consumers, offsets, partitions, replicas, and nodes)
- Run preferred replica election.
- Generate partition assignments with option to select brokers to use.

- Run reassignment of partition (based on generated assignments).
- Create a topic with optional topic configurations (Multiple Kafka cluster versions are supported).
- Delete a topic (only supported on 0.8.2+ and **delete.topic.enable=true** is set in broker configuration).
- Batch generate partition assignments for multiple topics with option to select brokers to use.
- Batch run reassignment of partitions for multiple topics.
- Add partitions to an existing topic.
- Update configurations for an existing topic.
- Optionally enable JMX polling for broker-level and topic-level metrics.
- Optionally filter out consumers that do not have ids/ owner / & offsets/ directories in ZooKeeper.

1.4.14 KrbServer and LdapServer

1.4.14.1 KrbServer and LdapServer Principles

Overview

To manage the access control permissions on data and resources in a cluster, it is recommended that the cluster be installed in security mode. In security mode, a client application must be authenticated and a secure session must be established before the application accesses any resource in the cluster. MRS uses KrbServer to provide Kerberos authentication for all components, implementing a reliable authentication mechanism.

LdapServer supports Lightweight Directory Access Protocol (LDAP) and provides the capability of storing user and user group data for Kerberos authentication.

Architecture

The security authentication function for user login depends on Kerberos and LDAP.

Figure 1-69 Security authentication architecture

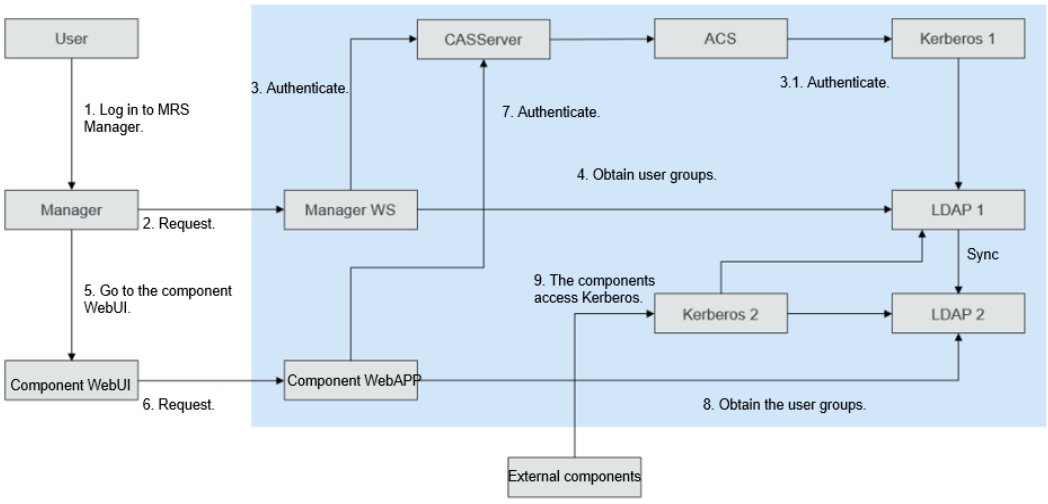


Figure 1-69 includes three scenarios:

- Logging in to the MRS Manager Web UI
  The authentication architecture includes steps 1, 2, 3, and 4.
- Logging in to a component web UI
  The authentication architecture includes steps 5, 6, 7, and 8.
- Accessing between components
  The authentication architecture includes step 9.

Table 1-14 Key modules

Connection Name	Description
Manager	Cluster Manager
Manager WS	WebBrowser
Kerberos1	KrbServer (management plane) service deployed in MRS Manager, that is, OMS Kerberos
Kerberos2	KrbServer (service plane) service deployed in the cluster
LDAP1	LdapServer (management plane) service deployed in MRS Manager, that is, OMS LDAP
LDAP2	LdapServer (service plane) service deployed in the cluster

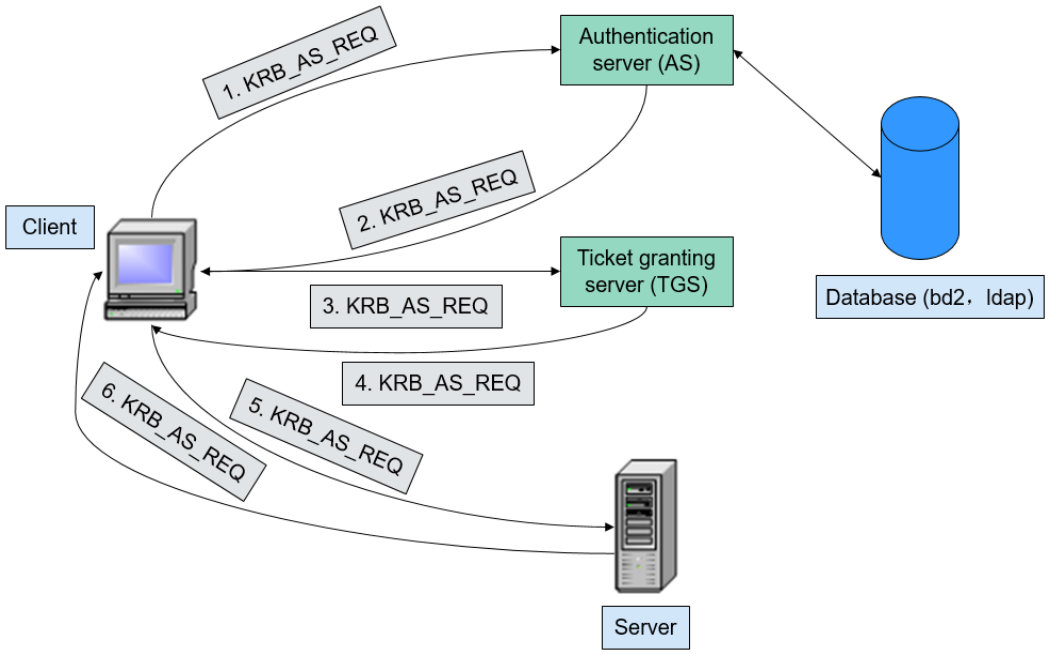
Data operation mode of Kerberos1 in LDAP: The active and standby instances of LDAP1 and the two standby instances of LDAP2 can be accessed in load balancing mode. Data write operations can be performed only in the active LDAP1 instance. Data read operations can be performed in LDAP1 or LDAP2.

Data operation mode of Kerberos2 in LDAP: Data read operations can be performed in LDAP1 and LDAP2. Data write operations can be performed only in the active LDAP1 instance.

Principle

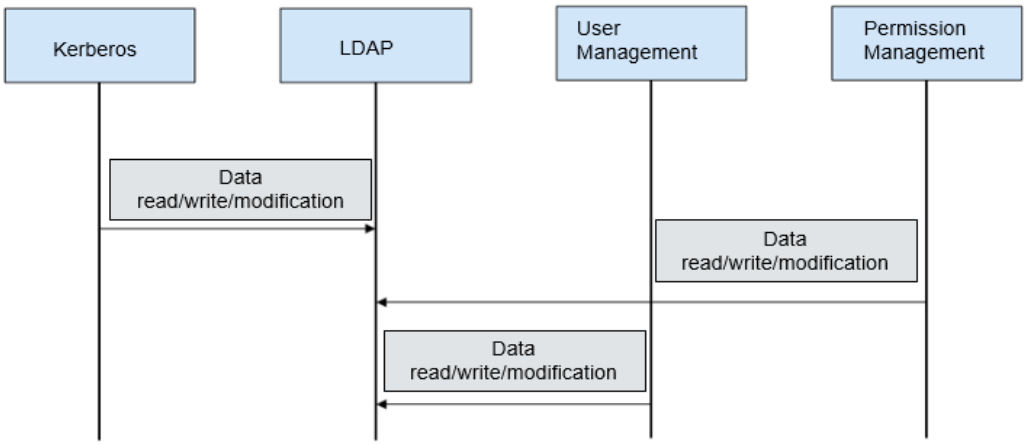
Kerberos authentication

Figure 1-70 Authentication process



LDAP data read and write

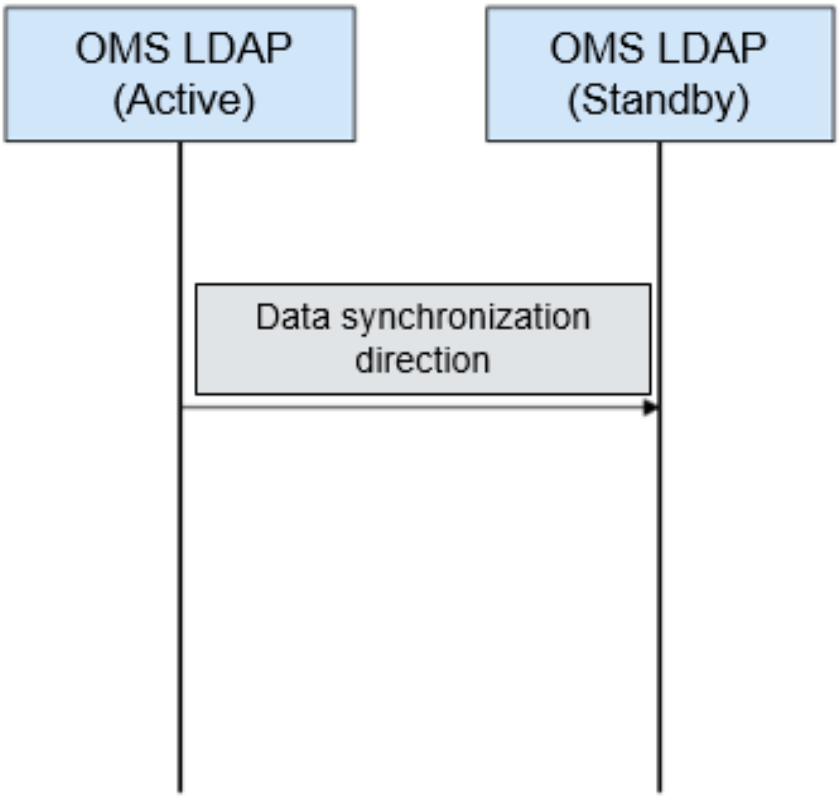
Figure 1-71 Data modification process



LDAP data synchronization

- OMS LDAP data synchronization before cluster installation

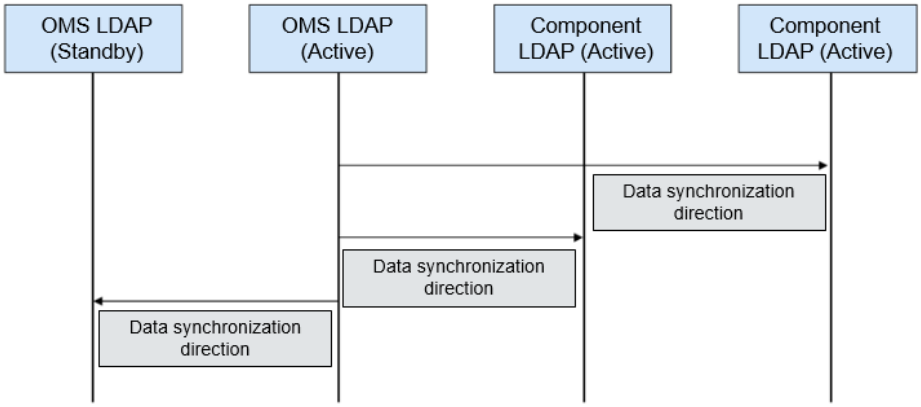
Figure 1-72 OMS LDAP data synchronization



Data synchronization direction before cluster installation: Data is synchronized from the active OMS LDAP to the standby OMS LDAP.

- LDAP data synchronization after cluster installation

Figure 1-73 LDAP data synchronization



Data synchronization direction after cluster installation: Data is synchronized from the active OMS LDAP to the standby OMS LDAP, standby component LDAP, and standby component LDAP.

1.4.14.2 KrbServer and LdapServer Enhanced Open Source Features

Enhanced open-source features of KrbServer and LdapServer: intra-cluster service authentication

In an MRS cluster that uses the security mode, mutual access between services is implemented based on the Kerberos security architecture. When a service (such as HDFS) in the cluster is to be started, the corresponding sessionkey (keytab, used for identity authentication of the application) is obtained from Kerberos. If another service (such as YARN) needs to access HDFS and add, delete, modify, or query data in HDFS, the corresponding TGT and ST must be obtained for secure access.

Enhanced Open-Source Features of KrbServer and LdapServer: Application Development Authentication

MRS components provide application development interfaces for customers or upper-layer service product clusters. During application development, a cluster in security mode provides specified application development authentication interfaces to implement application security authentication and access. For example, the UserGroupInformation class provided by the hadoop-common API provides multiple security authentication APIs.

- **setConfiguration()** is used to obtain related configuration and set parameters such as global variables.
- **loginUserFromKeytab():** is used to obtain TGT interfaces.

Enhanced Open-Source Features of KrbServer and LdapServer: Cross-System Mutual Trust

MRS provides the mutual trust function between two Managers to implement data read and write operations between systems.

1.4.15 Kudu

Kudu is a columnar storage manager developed for the Apache Hadoop platform. Kudu shares the common technical properties of Hadoop ecosystem applications: it runs on commodity hardware, is horizontally scalable, and supports highly available operation.

Kudu's design has the following benefits:

- Fast processing of OLAP workloads
- Integration with MapReduce, Spark and other Hadoop ecosystem components
- Tight integration with Apache Impala, making it a good, mutable alternative to using HDFS with Apache Parquet
- Strong but flexible consistency model, allowing you to choose consistency requirements on a per-request basis, including the option for strict-serializable consistency
- Strong performance for running sequential and random workloads simultaneously

- Easy to manage
- High availability Tablet Servers and Masters use the Raft Consensus Algorithm, which ensures that as long as more than half the total number of replicas is available, the tablet is available for reads and writes. For example, if 2 out of 3 replicas or 3 out of 5 replicas are available, the tablet is available. Reads can be serviced by read-only follower tablets, even in the event of a leader tablet failure.
- Structured data model

By combining all of these properties, Kudu targets support for families of applications that are difficult or impossible to implement on current generation Hadoop storage technologies.

A few examples of applications for which Kudu is a great solution are:

- Reporting applications where newly-arrived data needs to be immediately available for end users
- Time-series applications that must simultaneously support queries across large amounts of historic data and granular queries about an individual entity that must return very quickly
- Applications that use predictive models to make real-time decisions with periodic refreshes of the predictive model based on all historic data

1.4.16 Loader

1.4.16.1 Loader Basic Principles

Loader is developed based on the open source Sqoop component. It is used to exchange data and files between MRS and relational databases and file systems. Loader can import data from relational databases or file servers to the HDFS and HBase components, or export data from HDFS and HBase to relational databases or file servers.

A Loader model consists of Loader Client and Loader Server, as shown in **Figure 1-74**.

Figure 1-74 Loader model

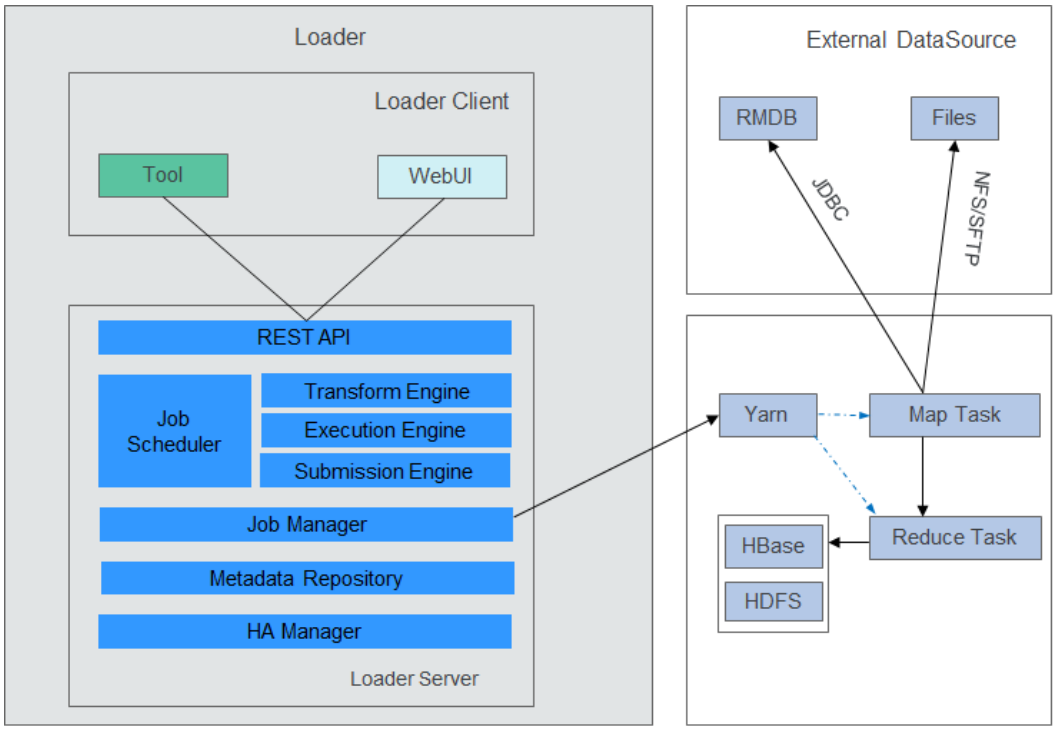


Table 1-15 describes the functions of each module shown in the preceding figure.

Table 1-15 Components of the Loader model

Module	Description
Loader Client	Loader client. It provides two interfaces: web UI and CLI.
Loader Server	Loader server. It processes operation requests sent from the client, manages connectors and metadata, submits MapReduce jobs, and monitors MapReduce job status.
REST API	It provides a Representational State Transfer (RESTful) APIs (HTTP + JSON) to process the operation requests sent from the client.
Job Scheduler	Simple job scheduler. It periodically executes Loader jobs.
Transform Engine	Data transformation engine. It supports field combination, string cutting, and string reverse.
Execution Engine	Loader job execution engine. It executes Loader jobs in MapReduce manner.
Submission Engine	Loader job submission engine. It submits Loader jobs to MapReduce.
Job Manager	It manages Loader jobs, including creating, querying, updating, deleting, activating, deactivating, starting, and stopping jobs.

Module	Description
Metadata Repository	Metadata repository. It stores and manages data about Loader connectors, transformation procedures, and jobs.
HA Manager	It manages the active/standby status of Loader Server processes. The Loader Server has two nodes that are deployed in active/standby mode.

Loader imports or exports jobs in parallel using MapReduce jobs. Some job import or export may involve only the Map operations, while some may involve both Map and Reduce operations.

Loader implements fault tolerance using MapReduce. Jobs can be rescheduled upon a job execution failure.

- **Importing data to HBase**

  When the Map operation is performed for MapReduce jobs, Loader obtains data from an external data source.

  When a Reduce operation is performed for a MapReduce job, Loader enables the same number of Reduce tasks based on the number of Regions. The Reduce tasks receive data from Map tasks, generate HFiles by Region, and store the HFiles in a temporary directory of HDFS.

  When a MapReduce job is submitted, Loader migrates HFiles from the temporary directory to the HBase directory.

- **Importing Data to HDFS**

  When a Map operation is performed for a MapReduce job, Loader obtains data from an external data source and exports the data to a temporary directory (named *export directory*-**ldtmp**).

  When a MapReduce job is submitted, Loader migrates data from the temporary directory to the output directory.

- **Exporting data to a relational database**

  When a Map operation is performed for a MapReduce job, Loader obtains data from HDFS or HBase and inserts the data to a temporary table (Staging Table) through the Java DataBase Connectivity (JDBC) API.

  When a MapReduce job is submitted, Loader migrates data from the temporary table to a formal table.

- **Exporting data to a file system**

  When a Map operation is performed for a MapReduce job, Loader obtains data from HDFS or HBase and writes the data to a temporary directory of the file server.

  When a MapReduce job is submitted, Loader migrates data from the temporary directory to a formal directory.

For details about the Loader architecture and principles, see **https://sqoop.apache.org/docs/1.99.3/index.html**.

1.4.16.2 Relationship Between Loader and Other Components

The components that interact with Loader include HDFS, HBase, MapReduce, and ZooKeeper. Loader works as a client to use certain functions of these components, such as storing data to HDFS and HBase and reading data from HDFS and HBase tables. In addition, Loader functions as an MapReduce client to import or export data.

1.4.16.3 Loader Enhanced Open Source Features

Loader Enhanced Open-Source Feature: Data Import and Export

Loader is developed based on Sqoop. In addition to the Sqoop functions, Loader has the following enhanced features:

- Provides data conversion functions.
- Supports GUI-based configuration conversion.
- Imports data from an SFTP/FTP server to HDFS/OBS.
- Imports data from an SFTP/FTP server to an HBase table.
- Imports data from an SFTP/FTP server to a Phoenix table.
- Imports data from an SFTP/FTP server to a Hive table.
- Exports data from HDFS/OBS to an SFTP/FTP server.
- Exports data from an HBase table to an SFTP/FTP server.
- Exports data from a Phoenix table to an SFTP/FTP server.
- Imports data from a relational database to an HBase table.
- Imports data from a relational database to a Phoenix table.
- Imports data from a relational database to a Hive table.
- Exports data from an HBase table to a relational database.
- Exports data from a Phoenix table to a relational database.
- Imports data from an Oracle partitioned table to HDFS/OBS.
- Imports data from an Oracle partitioned table to an HBase table.
- Imports data from an Oracle partitioned table to a Phoenix table.
- Imports data from an Oracle partitioned table to a Hive table.
- Exports data from HDFS/OBS to an Oracle partitioned table.
- Exports data from HBase to an Oracle partitioned table.
- Exports data from a Phoenix table to an Oracle partitioned table.
- Imports data from HDFS to an HBase table, a Phoenix table, and a Hive table in the same cluster.
- Exports data from an HBase table and a Phoenix table to HDFS/OBS in the same cluster.
- Imports data to an HBase table and a Phoenix table by using **bulkload** or **put list**.
- Imports all types of files from an SFTP/FTP server to HDFS. The open source component Sqoop can import only text files.
- Exports all types of files from HDFS/OBS to an SFTP server. The open source component Sqoop can export only text files and SequenceFile files.

- Supports file coding format conversion during file import and export. The supported coding formats include all formats supported by Java Development Kit (JDK).
- Retains the original directory structure and file names during file import and export.
- Supports file combination during file import and export. For example, if a large number of files are to be imported, these files can be combined into *n* files (*n* can be configured).
- Supports file filtering during file import and export. The filtering rules support wildcards and regular expressions.
- Supports batch import and export of ETL tasks.
- Supports query by page and key word and group management of ETL tasks.
- Provides floating IP addresses for external components.

1.4.17 Manager

1.4.17.1 Manager Basic Principles

Overview

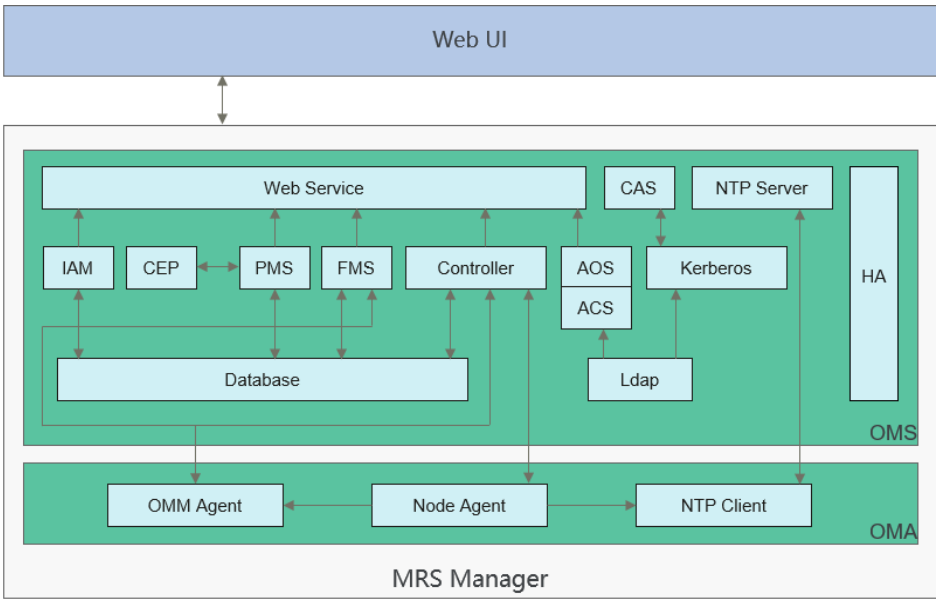
Manager is the O&M management system of MRS and provides unified cluster management capabilities for services deployed in clusters.

Manager provides functions such as performance monitoring, alarms, user management, permission management, auditing, service management, health check, and log collection.

Architecture

Figure 1-75 shows the overall logical architecture of FusionInsight Manager.

Figure 1-75 Manager logical architecture



Manager consists of OMS and OMA.

- OMS: serves as management node in the O&M system. There are two OMS nodes deployed in active/standby mode.
- OMA: managed node in the O&M system. Generally, there are multiple OMA nodes.

Figure 1-75 describes the modules shown in **Table 1-16**.

Table 1-16 Service module description

Module	Description
Web Service	A web service deployed under Tomcat, providing HTTPS API of Manager. It is used to access Manager through the web browser. In addition, it provides the northbound access capability based on the Syslog and SNMP protocols.
OMS	Management node of the O&M system. Generally, there are two OMS nodes that work in active/standby mode.
OMA	Managed node in the O&M system. Generally, there are multiple OMA nodes.
Controller	The control center of Manager. It can converge information from all nodes in the cluster and display it to the MRS administrators, as well as receive from the MRS cluster administrators, and synchronize information to all nodes in the cluster according to the operation instruction range. Control process of Manager. It implements various management actions: 1. The web service delivers various management actions (such as installation, service startup and stop, and configuration modification) to Controller. 2. Controller decomposes the command and delivers the action to each Node Agent, for example, starting a service involves multiple roles and instances. 3. Controller is responsible for monitoring the implementation of each action.
Node Agent	Node Agent exists on each cluster node and is an enabler of Manager on a single node. • Node Agent represents all the components deployed on the node to interact with Controller, implementing convergence from multiple nodes of a cluster to a single node. • Node Agent enables Controller to perform all operations on the components deployed on the node. It allows Controller functions to be implemented. Node Agent sends heartbeat messages to Controller at an interval of 3 seconds. The interval cannot be configured.
IAM	Records audit logs. Each non-query operation on the Manager UI has a related audit log.

Module	Description
PMS	The performance monitoring module. It collects the performance monitoring data on each OMA and provides the query function.
CEP	Convergence function module. For example, the used disk space of all OMAs is collected as a performance indicator.
FMS	Alarm module. It collects and queries alarms on each OMA.
OMM Agent	Agent for performance monitoring and alarm reporting on the OMA. It collects performance monitoring data and alarm data on Agent Node.
CAS	Unified authentication center. When a user logs in to the web service, CAS authenticates the login. The browser automatically redirects the user to the CAS through URLs.
AOS	Permission management module. It manages the permissions of users and user groups.
ACS	User and user group management module. It manages users and user groups to which users belong.
Kerberos	LDAP is deployed in OMS and a cluster, respectively. ● OMS Kerberos provides the single sign-on (SSO) and authentication between Controller and Node Agent. ● Kerberos in the cluster provides the user security authentication function for components. The service name is **KrbServer**, which contains two role instances: – KerberosServer: is an authentication server that provides security authentication for MRS. – KerberosAdmin: manages processes of Kerberos users.
Ldap	LDAP is deployed in OMS and a cluster, respectively. ● OMS LDAP provides data storage for user authentication. ● The LDAP in the cluster functions as the backup of the OMS LDAP. The service name is **LdapServer** and the role instance is **SlapdServer**.
Database	Manager database used to store logs and alarms.
HA	HA management module that manages the active and standby OMSs.
NTP Server NTP Client	It synchronizes the system clock of each node in the cluster.

1.4.17.2 Manager Key Features

Key Feature: Unified Alarm Monitoring

Manager provides the visualized and convenient alarm monitoring function. Users can quickly obtain key cluster performance indicators, evaluate cluster health status, customize performance indicator display, and convert indicators to alarms. Manager can monitor the running status of all components and report alarms in real time when faults occur. The online help on the GUI allows you to view performance counters and alarm clearance methods to quickly rectify faults.

Key Feature: Unified User Permission Management

Manager provides permission management of components in a unified manner.

Manager introduces the concept of role and uses role-based access control (RBAC) to manage system permissions. It centrally displays and manages scattered permission functions of each component in the system and organizes the permissions of each component in the form of permission sets (roles) to form a unified system permission concept. By doing so, common users cannot obtain internal permission management details, and permissions become easy for the MRS cluster administrators to manage, greatly facilitating permission management and improving user experience.

Key Feature: SSO

Single sign-on (SSO) is provided between the Manager web UI and component web UI as well as for integration between MRS and third-party systems.

This function centrally manages and authenticates Manager users and component users. The entire system uses LDAP to manage users and uses Kerberos for authentication. A set of Kerberos and LDAP management mechanisms are used between the OMS and components. SSO (including single sign-on and single sign-out) is implemented through CAS. With SSO, users can easily switch tasks between the Manager web UI, component web UIs, and third-party systems, without switching to another user.

NOTE

- To ensure security, the CAS Server can retain a ticket-granting ticket (TGT) used by a user only for 20 minutes.
- If a user does not perform any operation on the page (including on the Manager web UI and component web UIs) within 20 minutes, the page is automatically locked.

Key Feature: Automatic Health Check and Inspection

Manager provides users with automatic inspection on system running environments and helps users check and audit system running health by one click, ensuring correct system running and lowering system operation and maintenance costs. After viewing inspection results, you can export reports for archiving and fault analysis.

Key Feature: Tenant Management

Manager introduces the multi-tenant concept. The CPU, memory, and disk resources of a cluster can be integrated into a set. The set is called a tenant. A mode involving different tenants is called multi-tenant mode.

Manager provides the multi-tenant function, supports a level-based tenant model and allows tenants to be added and deleted dynamically, achieving resource isolation. As a result, it can dynamically manage and configure the computing resources and the storage resources of tenants.

- The computing resources indicate tenants' Yarn task queue resources. The task queue quota can be modified, and the task queue usage status and statistics can be viewed.
- The storage resources can be stored on HDFS. You can add and delete the HDFS storage directories of tenants, and set the quotas of file quantity and the storage space of the directories.

As a unified tenant management platform of MRS, MRS Manager allows users to create and manage tenants in clusters based on service requirements.

- Roles, computing resources, and storage resources are automatically created when tenants are created. By default, all permissions of the new computing resources and storage resources are allocated to a tenant's roles.
- After you have modified the tenant's computing or storage resources, permissions of the tenant's roles are automatically updated.

Manager also provides the multi-instance function so that users can use the HBase, Hive, or Spark alone in the resource control and service isolation scenario. The multi-instance function is disabled by default and can be manually enabled.

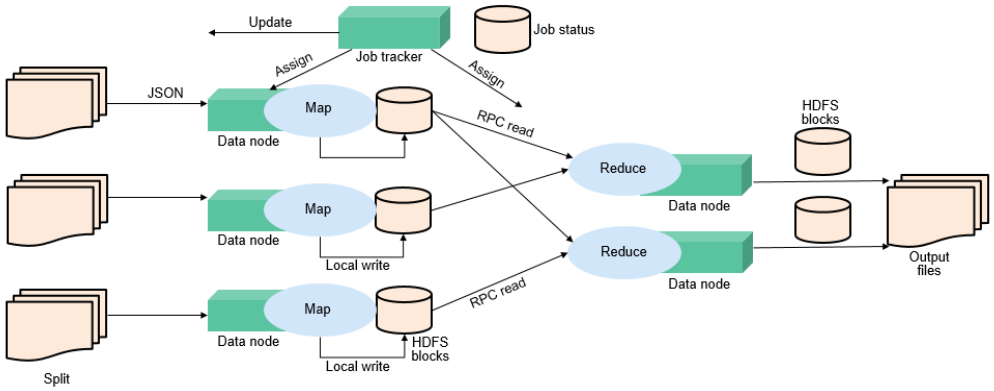
1.4.18 MapReduce

1.4.18.1 MapReduce Basic Principles

MapReduce is the core of Hadoop. As a software architecture proposed by Google, MapReduce is used for parallel computing of large-scale datasets (larger than 1 TB). The concepts "Map" and "Reduce" and their main thoughts are borrowed from functional programming language and also borrowed from the features of vector programming language.

Current software implementation is as follows: Specify a Map function to map a series of key-value pairs into a new series of key-value pairs, and specify a Reduce function to ensure that all values in the mapped key-value pairs share the same key.

Figure 1-76 Distributed batch processing engine



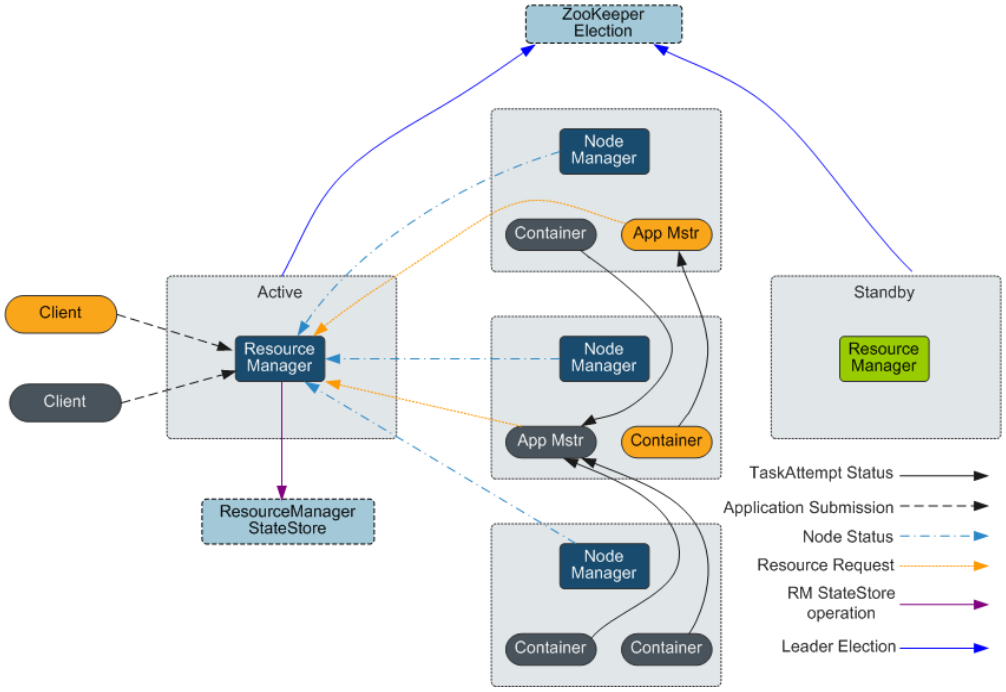
MapReduce is a software framework for processing large datasets in parallel. The root of MapReduce is the Map and Reduce functions in functional programming. The Map function accepts a group of data and transforms it into a key-value pair list. Each element in the input domain corresponds to a key-value pair. The Reduce function accepts the list generated by the Map function, and then shrinks the key-value pair list based on the keys. MapReduce divides a task into multiple parts and allocates them to different devices for processing. In this way, the task can be finished in a distributed environment instead of a single powerful server.

For more information, see **MapReduce Tutorial**.

MapReduce structure

As shown in **Figure 1-77**, MapReduce is integrated into YARN through the Client and ApplicationMaster interfaces of YARN, and uses YARN to apply for computing resources.

Figure 1-77 Basic architecture of Apache YARN and MapReduce



1.4.18.2 Relationship Between MapReduce and Other Components

Relationship Between MapReduce and HDFS

- HDFS features high fault tolerance and high throughput, and can be deployed on low-cost hardware for storing data of applications with massive data sets.
- MapReduce is a programming model used for parallel computation of large data sets (larger than 1 TB). Data computed by MapReduce comes from multiple data sources, such as Local FileSystem, HDFS, and databases. Most data comes from the HDFS. The high throughput of HDFS can be used to read massive data. After being computed, data can be stored in HDFS.

Relationship Between MapReduce and Yarn

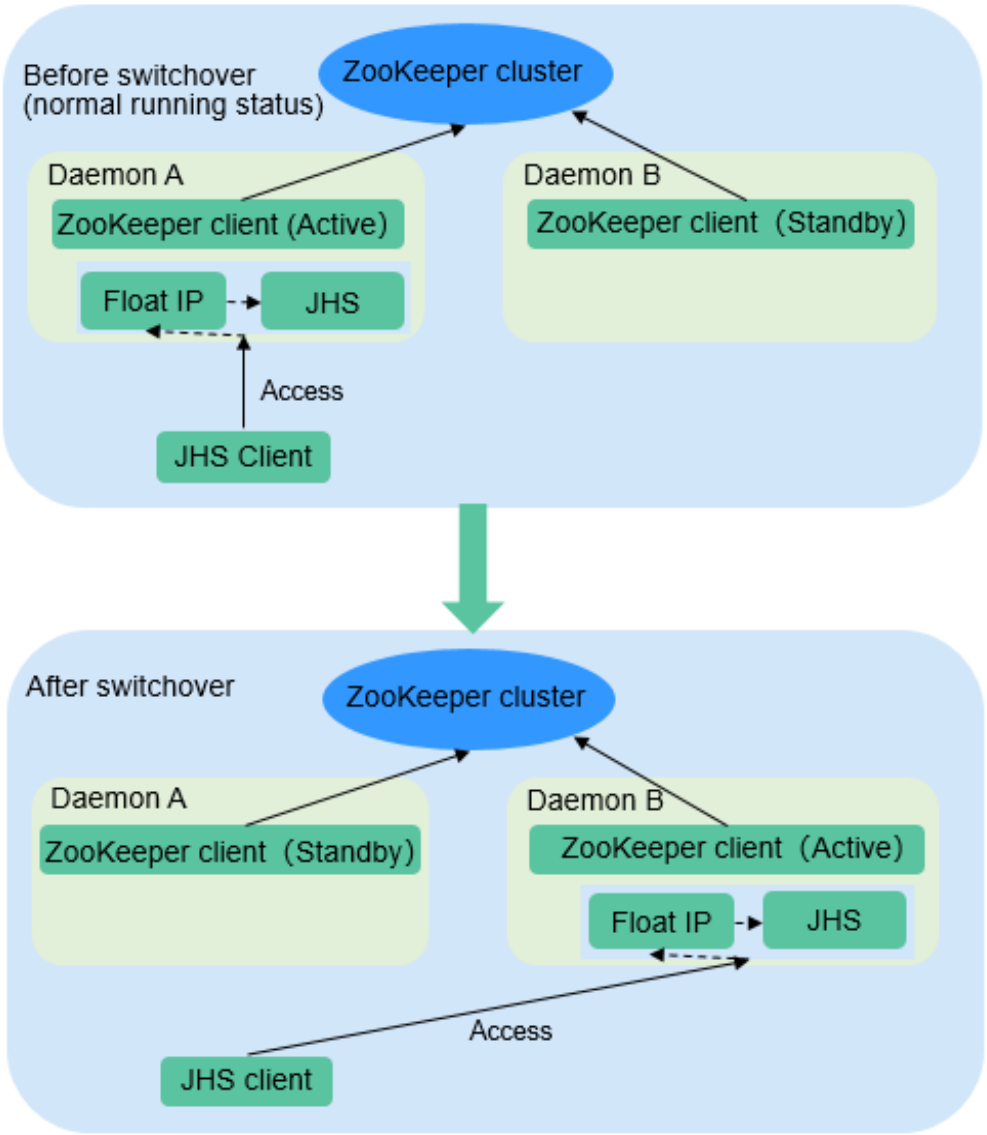
MapReduce is a computing framework running on Yarn, which is used for batch processing. MRv1 is implemented based on MapReduce in Hadoop 1.0, which is composed of programming models (new and old programming APIs), running environment (JobTracker and TaskTracker), and data processing engine (MapTask and ReduceTask). This framework is still weak in scalability, fault tolerance (JobTracker SPOF), and compatibility with multiple frameworks. (Currently, only the MapReduce computing framework is supported.) MRv2 is implemented based on MapReduce in Hadoop 2.0. The source code reuses MRv1 programming models and data processing engine implementation, and the running environment is composed of ResourceManager and ApplicationMaster. ResourceManager is a brand new resource manager system, and ApplicationMaster is responsible for cutting MapReduce job data, assigning tasks, applying for resources, scheduling tasks, and tolerating faults.

1.4.18.3 MapReduce Enhanced Open Source Features

MapReduce Enhanced Open-Source Feature: JobHistoryServer HA

JobHistoryServer (JHS) is the server used to view historical MapReduce task information. Currently, the open source JHS supports only single-instance services. JHS HA can solve the problem that an application fails to access the MapReduce API when SPOFs occur on the JHS, which causes the application fails to be executed. This greatly improves the high availability of the MapReduce service.

Figure 1-78 Status transition of the JobHistoryServer HA active/standby switchover



JobHistoryServer High Availability

- ZooKeeper is used to implement active/standby election and switchover.
- JHS uses the floating IP address to provide services externally.
- Both the JHS single-instance and HA deployment modes are supported.
- Only one node starts the JHS process at a time point to prevent multiple JHS operations from processing the same file.
- You can perform scale-out, scale-in, instance migration, upgrade, and health check.

Enhanced Open Source Feature: Improving MapReduce Performance by Optimizing the Merge/Sort Process in Specific Scenarios

The figure below shows the workflow of a MapReduce task.

Figure 1-79 MapReduce job

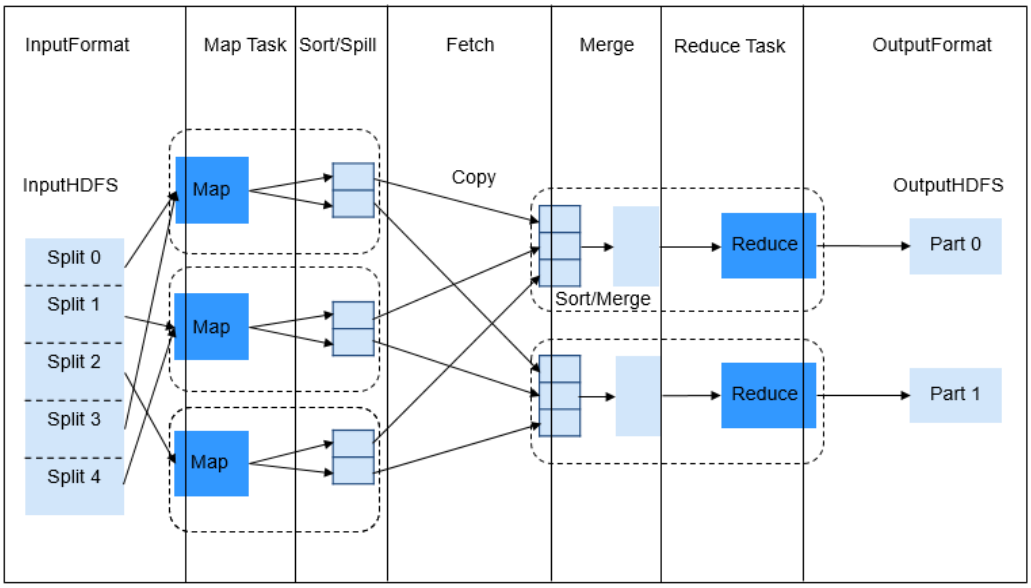
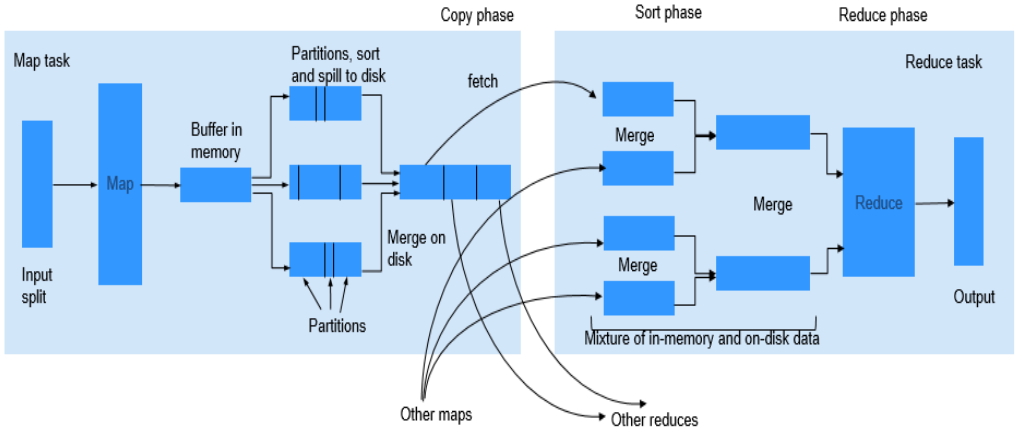


Figure 1-80 MapReduce job execution flow



The Reduce process is divided into three different steps: Copy, Sort (actually supposed to be called Merge), and Reduce. In Copy phase, Reducer tries to fetch the output of Maps from NodeManagers and store it on Reducer either in memory or on disk. Shuffle (Sort and Merge) phase then begins. All the fetched map outputs are being sorted, and segments from different map outputs are merged before being sent to Reducer. When a job has a large number of maps to be processed, the shuffle process is time-consuming. For specific tasks (for example, SQL tasks such as hash join and hash aggregation), sorting is not mandatory during the shuffle process. However, the sorting is required by default in the shuffle process.

This feature is enhanced by using the MapReduce API, which can automatically close the Sort process for such tasks. When the sorting is disabled, the API directly merges the fetched Maps output data and sends the data to Reducer. This greatly saves time, and significantly improves the efficiency of SQL tasks.

Enhanced Open Source Feature: Small Log File Problem Solved After Optimization of MR History Server

After the job running on Yarn is executed, NodeManager uses LogAggregationService to collect and send generated logs to HDFS and deletes them from the local file system. After the logs are stored to HDFS, they are managed by MR HistoryServer. LogAggregationService will merge local logs generated by containers to a log file and upload it to the HDFS, reducing the number of log files to some extent. However, in a large-scale and busy cluster, there will be excessive log files on HDFS after long-term running.

For example, if there are 20 nodes, about 18 million log files are generated within the default clean-up period (15 days), which occupy about 18 GB of the memory of a NameNode and slow down the HDFS system response.

Only the reading and deletion are required for files stored on HDFS. Therefore, Hadoop Archives can be used to periodically archive the directory of collected log files.

Archiving Logs

The AggregatedLogArchiveService module is added to MR HistoryServer to periodically check the number of files in the log directory. When the number of files reaches the threshold, AggregatedLogArchiveService starts an archiving task to archive log files. After archiving, it deletes the original log files to reduce log files on HDFS.

Cleaning Archived Logs

Hadoop Archives does not support deletion in archived files. Therefore, the entire archive log package must be deleted upon log clean-up. The latest log generation time is obtained by modifying the AggregatedLogDeletionService module. If all log files meet the clean-up requirements, the archive log package can be deleted.

Browsing Archived Logs

Hadoop Archives allows URI-based access to file content in the archive log package. Therefore, if MR History Server detects that the original log files do not exist during file browsing, it directly redirects the URI to the archive log package to access the archived log file.

NOTE

- This function invokes Hadoop Archives of HDFS for log archiving. Because the execution of an archiving task by Hadoop Archives is to run an MR application. Therefore, after an archiving task is executed, an MR execution record is added.
- This function of archiving logs is based on the log collection function. Therefore, this function is valid only when the log collection function is enabled.

1.4.19 Oozie

1.4.19.1 Oozie Basic Principles

Introduction to Oozie

Oozie is an open-source workflow engine that is used to schedule and coordinate Hadoop jobs.

Architecture

The Oozie engine is a web application integrated into Tomcat by default. Oozie uses PostgreSQL databases.

Oozie provides an Ext-based web console, through which users can view and monitor Oozie workflows. Oozie provides an external REST web service API for the Oozie client to control workflows (such as starting and stopping operations), and orchestrate and run Hadoop MapReduce tasks. For details, see Figure 1-81.

Figure 1-81 Oozie architecture

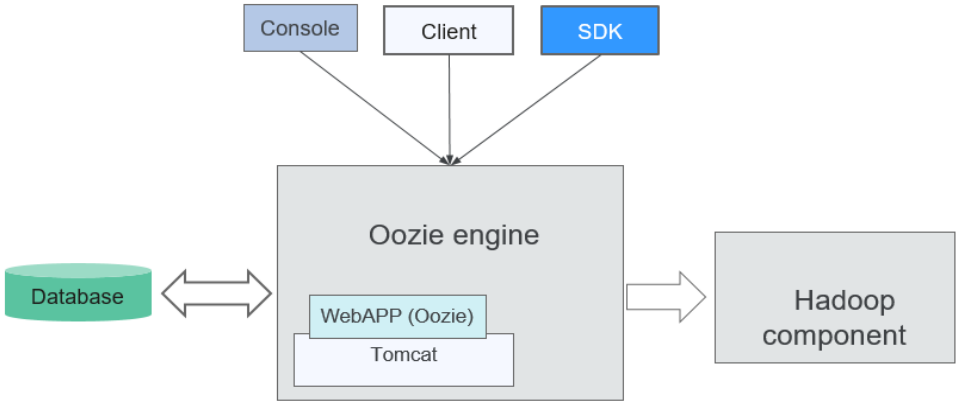


Table 1-17 describes the functions of each module shown in Figure 1-81.

Table 1-17 Architecture description

Connection Name	Description
Console	Allows users to view and monitor Oozie workflows.
Client	Controls workflows, including submitting, starting, running, planting, and restoring workflows, through APIs.
SDK	Is short for software development kit. An SDK is a set of development tools used by software engineers to establish applications for particular software packages, software frameworks, hardware platforms, and operating systems.
Database	PostgreSQL database

Connection Name	Description
WebApp (Oozie)	Functions as the Oozie server. It can be deployed on a built-in or an external Tomcat container. Information recorded by WebApp (Oozie) including logs is stored in the PostgreSQL database.
Tomcat	A free open-source web application server
Hadoop components	Underlying components, such as MapReduce and Hive, that execute the workflows orchestrated by Oozie.

Principle

Oozie is a workflow engine server that runs MapReduce workflows. It is also a Java web application running in a Tomcat container.

Oozie workflows are constructed using Hadoop Process Definition Language (HPDL). HPDL is an XML-defined language, similar to JBoss jBPM Process Definition Language (jPDL). An Oozie workflow consists of the Control Node and Action Node.

- Control Node controls workflow orchestration, such as **start**, **end**, **error**, **decision**, **fork**, and **join**.
- An Oozie workflow contains multiple Action Nodes, such as MapReduce and Java.

  All Action Nodes are deployed and run in Direct Acyclic Graph (DAG) mode. Therefore, Action Nodes run in direction. That is, the next Action Node can run only when the running of the previous Action Node ends. When one Action Node ends, the remote server calls back the Oozie interface. Then Oozie executes the next Action Node of workflow in the same manner until all Action Nodes are executed (execution failures are counted).

Oozie workflows provide various types of Action Nodes, such as MapReduce, Hadoop distributed file system (HDFS), Secure Shell (SSH), Java, and Oozie sub-flows, to support a wide range of business requirements.

1.4.19.2 Oozie Enhanced Open Source Features

Enhanced Open Source Feature: Improved Security

Provides roles of administrator and common users to support Oozie permission management.

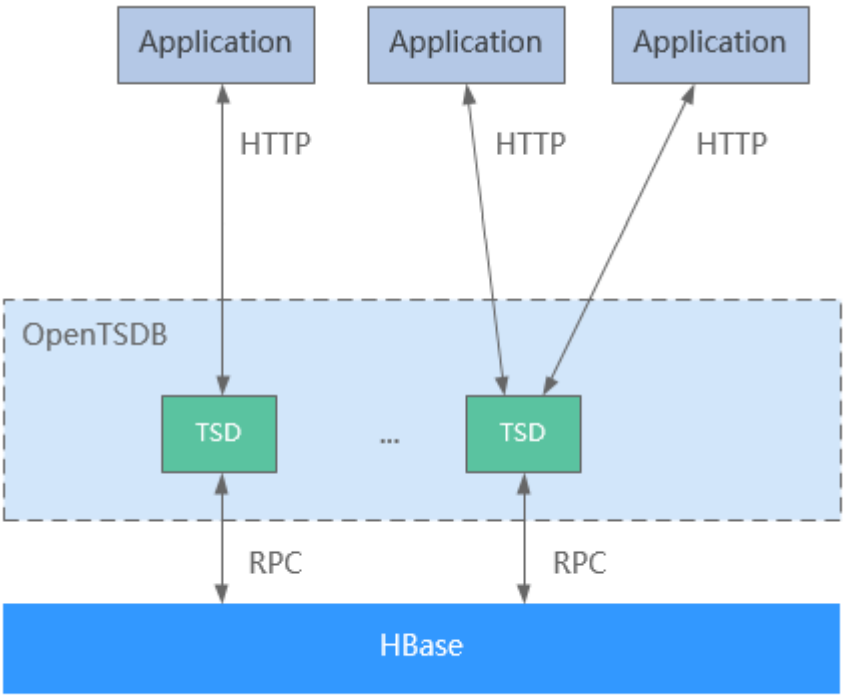
Supports single sign-on and sign-out, HTTPS access, and audit logs.

1.4.20 OpenTSDB

OpenTSDB is a distributed, scalable time series database based on HBase. OpenTSDB is designed to collect monitoring information of a large-scale cluster and implement second-level data query, eliminating the limitations of querying and storing massive amounts of monitoring data in common databases.

OpenTSDB consists of a Time Series Daemon (TSD) as well as a set of command line utilities. Interaction with OpenTSDB is primarily implemented by running one or more TSDs. Each TSD is independent. There is no master server and no shared state, so you can run as many TSDs as required to handle any load you throw at it. Each TSD uses HBase in a CloudTable cluster to store and retrieve time series data. The data schema is highly optimized for fast aggregations of similar time series to minimize storage space. TSD users never need to directly access the underlying storage. You can communicate with the TSD through an HTTP API. All communications happen on the same port (the TSD figures out the protocol of the client by looking at the first few bytes it receives).

Figure 1-82 OpenTSDB architecture



Application scenarios of OpenTSDB have the following features:

- The collected metrics have a unique value at a time point and do not have a complex structure or relationship.
- Monitoring metrics change with time.
- Like HBase, OpenTSDB features high throughput and good scalability.

OpenTSDB provides an HTTP based application programming interface to enable integration with external systems. Almost all OpenTSDB features are accessible via the API such as querying time series data, managing metadata, and storing data points. For details, visit **https://opentsdb.net/docs/build/html/api_http/index.html**.

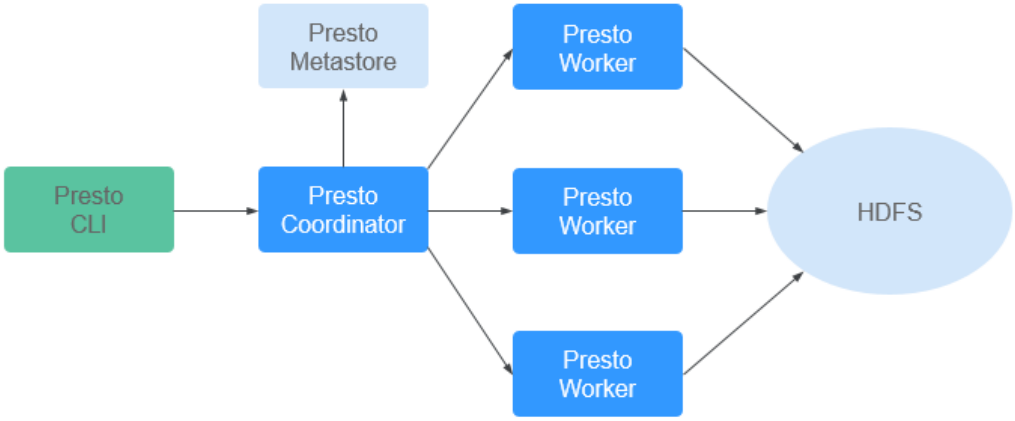
1.4.21 Presto

Presto is an open source SQL query engine for running interactive analytic queries against data sources of all sizes. It applies to massive structured/semi-structured

data analysis, massive multi-dimensional data aggregation/report, ETL, ad-hoc queries, and more scenarios.

Presto allows querying data where it lives, including HDFS, Hive, HBase, Cassandra, relational databases or even proprietary data stores. A Presto query can combine different data sources to perform data analysis across the data sources.

Figure 1-83 Presto architecture



Presto runs in a cluster in distributed mode and contains one coordinator and multiple worker processes. Query requests are submitted from clients (for example, CLI) to the coordinator. The coordinator parses SQL statements, generates execution plans, and distributes the plans to multiple worker processes for execution.

For details about Presto, visit **https://prestodb.github.io/** or **https://prestosql.io/**.

Multiple Presto Instances

MRS supports the installation of multiple Presto instances for a large-scale cluster by default. That is, multiple Worker instances, such as Worker1, Worker2, and Worker3, are installed on a Core/Task node. Multiple Worker instances interact with the Coordinator to execute computing tasks, greatly improving node resource utilization and computing efficiency.

Presto multi-instance applies only to the Arm architecture. Currently, a single node supports a maximum of four instances.

For more Presto deployment information, see **https://prestodb.io/docs/current/installation/deployment.html** or **https://trino.io/docs/current/installation/deployment.html**.

1.4.22 Ranger

1.4.22.1 Ranger Basic Principles

Apache Ranger offers a centralized security management framework and supports unified authorization and auditing. It manages fine grained access

control over Hadoop and related components, such as Storm, HDFS, Hive, HBase, and Kafka. You can use the front-end web UI console provided by Ranger to configure policies to control users' access to these components.

Figure 1-84 shows the Ranger architecture.

Figure 1-84 Ranger structure

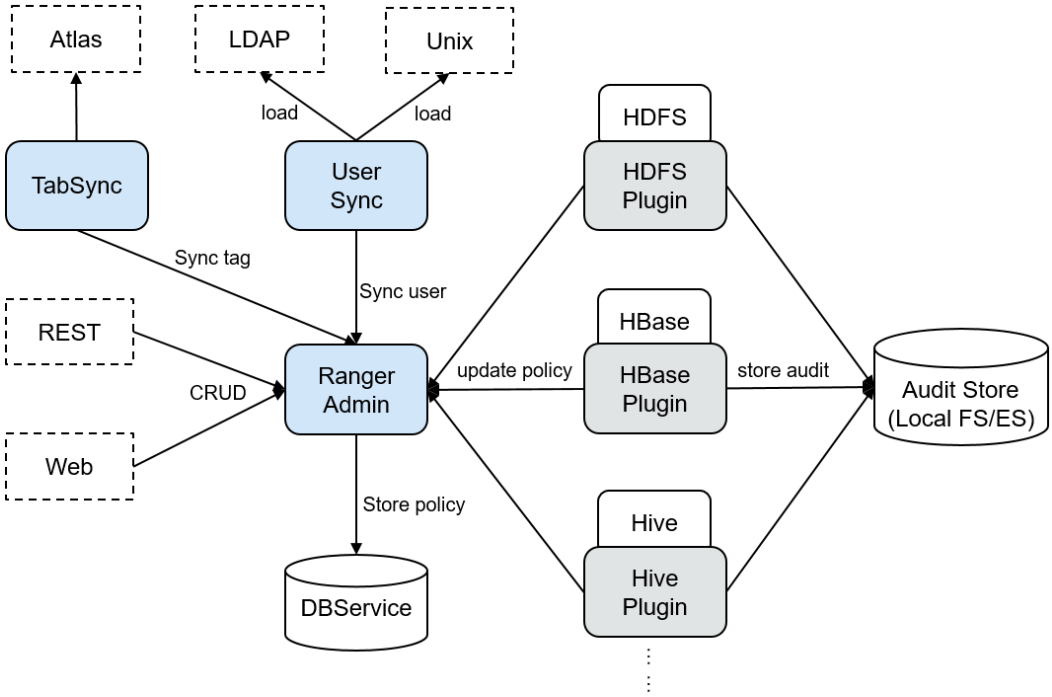


Table 1-18 Architecture description

Connection Name	Description
RangerAdmin	Provides a WebUI and RESTful API to manage policies, users, and auditing.
UserSync	Periodically synchronizes user and user group information from an external system and writes the information to RangerAdmin.
TagSync	Periodically synchronizes tag information from the external Atlas service and writes the tag information to RangerAdmin.

Ranger Principles

- Ranger Plugins

  Ranger provides policy-based access control (PBAC) plug-ins to replace the original authentication plug-ins of the components. Ranger plug-ins are developed based on the authentication interface of the components. Users set permission policies for specified services on the Ranger WebUI. Ranger plug-ins periodically update policies from the RangerAdmin and caches them in the

local file of the component. When a client request needs to be authenticated, the Ranger plug-in matches the user carried in the request with the policy and then returns an accept or reject message.

- UserSync User Synchronization

  UserSync periodically synchronizes data from LDAP/Unix to RangerAdmin. In security mode, data is synchronized from LDAP. In non-security mode, data is synchronized from Unix. By default, the incremental synchronization mode is used. In each synchronization period, UserSync updates only new or modified users and user groups. When a user or user group is deleted, UserSync does not synchronize the change to RangerAdmin. That is, the user or user group is not deleted from the RangerAdmin. To improve performance, UserSync does not synchronize user groups to which no user belongs to RangerAdmin.

- Unified auditing

  Ranger plug-ins can record audit logs. Currently, audit logs can be stored in local files.

- High reliability

  Ranger supports two RangerAdmins working in active/active mode. Two RangerAdmins provide services at the same time. If either RangerAdmin is faulty, Ranger continues to work.

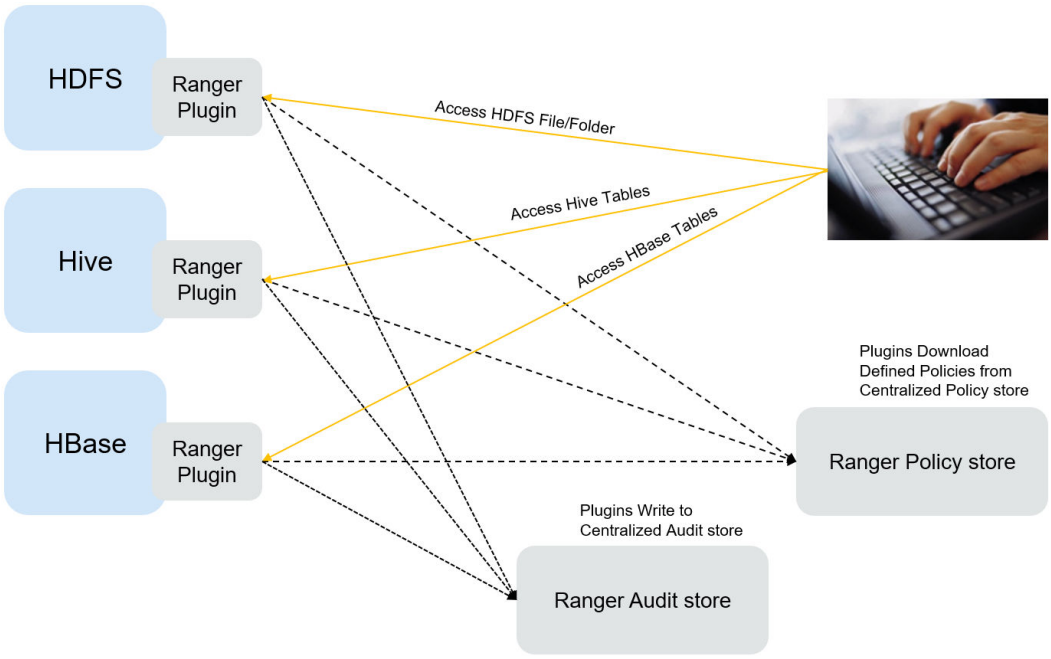
- High performance

  Ranger provides the Load-Balance capability. When a user accesses Ranger WebUI using a browser, the Load-Balance automatically selects the RangerAdmin with the lightest load to provide services.

1.4.22.2 Relationship Between Ranger and Other Components

Ranger provides PABC-based authentication plug-ins for components to run on their servers. Ranger currently supports authentication for the following components like HDFS, YARN, Hive, HBase, Kafka, Storm, and Spark2x. More components will be supported in the future.

Figure 1-85 Relationship Between Ranger and Other Components



1.4.23 Spark

1.4.23.1 Basic Principles of Spark

NOTE

The Spark component applies to versions earlier than MRS 3.x.

Description

Spark is an open source parallel data processing framework. It helps you to easily develop unified big data applications and perform offline processing, stream processing, and interactive analysis on data.

Spark provides a framework featuring fast computing, write, and interactive query. Spark has obvious advantages over Hadoop in terms of performance. Spark uses the in-memory computing mode to avoid I/O bottlenecks in scenarios where multiple tasks in a MapReduce workflow process the same dataset. Spark is implemented by using Scala programming language. Scala enables distributed datasets to be processed in a method that is the same as that of processing local data. In addition to interactive data analysis, Spark supports interactive data mining. Spark adopts in-memory computing, which facilitates iterative computing. By coincidence, iterative computing of the same data is a general problem facing data mining. In addition, Spark can run in Yarn clusters where Hadoop 2.0 is installed. The reason why Spark cannot only retain various features like MapReduce fault tolerance, data localization, and scalability but also ensure high performance and avoid busy disk I/Os is that a memory abstraction structure called Resilient Distributed Dataset (RDD) is created for Spark.

Original distributed memory abstraction, for example, key-value store and databases, supports small-granularity update of variable status. This requires

backup of data or log updates to ensure fault tolerance. Consequently, a large amount of I/O consumption is brought about to data-intensive workflows. For the RDD, it has only one set of restricted APIs and only supports large-granularity update, for example, map and join. In this way, Spark only needs to record the transformation operation logs generated during data establishment to ensure fault tolerance without recording a complete dataset. This data transformation link record is a source for tracing a data set. Generally, parallel applications apply the same computing process for a large dataset. Therefore, the limit to the mentioned large-granularity update is not large. As described in Spark theses, the RDD can function as multiple different computing frameworks, for example, programming models of MapReduce and Pregel. In addition, Spark allows you to explicitly make a data transformation process be persistent on hard disks. Data localization is implemented by allowing you to control data partitions based on the key value of each record. (An obvious advantage of this method is that two copies of data to be associated will be hashed in the same mode.) If memory usage exceeds the physical limit, Spark writes relatively large partitions into hard disks, thereby ensuring scalability.

Spark has the following features:

- Fast: The data processing speed of Spark is 10 to 100 times higher than that of MapReduce.
- Easy-to-use: Java, Scala, and Python can be used to simply and quickly compile parallel applications for processing massive amounts of data. Spark provides over 80 operators to help you compile parallel applications.
- Integration with Hadoop: Spark can directly run in a Hadoop cluster and read existing Hadoop data.

The Spark component of MRS has the following advantages:

- The Spark Streaming component of MRS supports real-time data processing rather than triggering as scheduled.
- The Spark component of MRS provides Structured Streaming and allows you to build streaming applications using the Dataset API. Spark supports exactly-once semantics and inner and outer joins for streams.
- The Spark component of MRS uses **pandas_udf** to replace the original user-defined functions (UDFs) in PySpark to process data, which reduces the processing duration by 60% to 90% (affected by specific operations).
- The Spark component of MRS also supports graph data processing and allows modeling using graphs during graph computing.
- Spark SQL of MRS is compatible with some Hive syntax (based on the 64 SQL statements of the Hive-Test-benchmark test set) and standard SQL syntax (based on the 99 SQL statements of the TPC-DS test set).

Architecture

Figure 1-86 describes the Spark architecture and **Table 1-19** lists the Spark modules.

Figure 1-86 Spark architecture

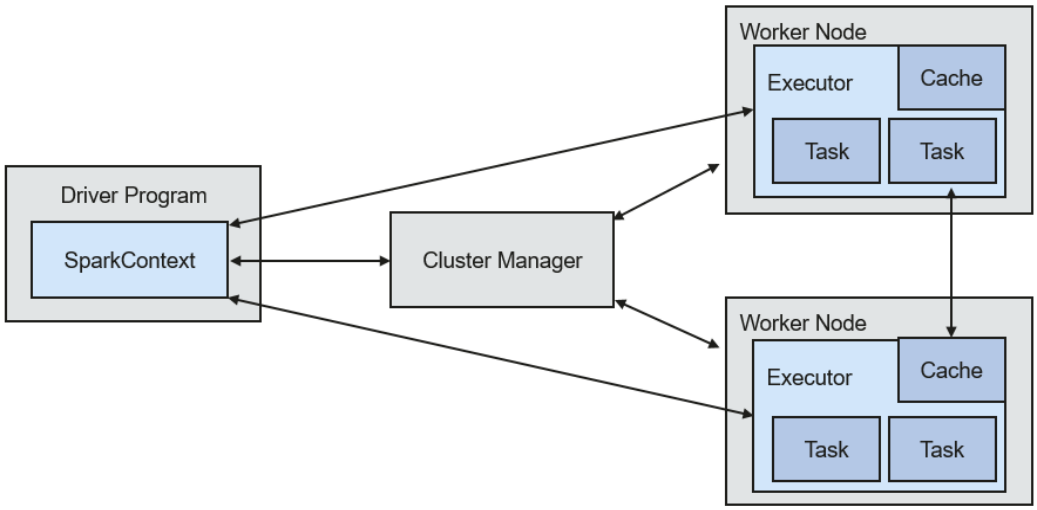


Table 1-19 Basic concepts

Module	Description
Cluster Manager	Cluster manager manages resources in the cluster. Spark supports multiple cluster managers, including Mesos, Yarn, and the Standalone cluster manager that is delivered with Spark.
Application	Spark application. It consists of one Driver Program and multiple executors.
Deploy Mode	Deployment in cluster or client mode. In cluster mode, the driver runs on a node inside the cluster. In client mode, the driver runs on the client (outside the cluster).
Driver Program	The main process of the Spark application. It runs the **main()** function of an application and creates SparkContext. It is used for parsing applications, generating stages, and scheduling tasks to executors. Usually, SparkContext represents Driver Program.
Executor	A process started on a Work Node. It is used to execute tasks, and manage and process the data used in applications. A Spark application usually contains multiple executors. Each executor receives commands from the driver and executes one or multiple tasks.
Worker Node	A node that starts and manages executors and resources in a cluster.
Job	A job consists of multiple concurrent tasks. One action operator (for example, a collect operator) maps to one job.
Stage	Each job consists of multiple stages. Each stage is a task set, which is separated by Directed Acyclic Graph (DAG).

Module	Description
Task	A task carries the computation unit of the service logics. It is the minimum working unit that can be executed on the Spark platform. An application can be divided into multiple tasks based on the execution plan and computation amount.

Spark Application Running Principle

Figure 1-87 shows the Spark application running architecture. The running process is as follows:

1. An application is running in the cluster as a collection of processes. Driver coordinates the running of the application.
2. To run an application, Driver connects to the cluster manager (such as Standalone, Mesos, and Yarn) to apply for the executor resources, and start ExecutorBackend. The cluster manager schedules resources between different applications. Driver schedules DAGs, divides stages, and generates tasks for the application at the same time.
3. Then, Spark sends the codes of the application (the codes transferred to **SparkContext**, which is defined by JAR or Python) to an executor.
4. After all tasks are finished, the running of the user application is stopped.

Figure 1-87 Spark application running architecture

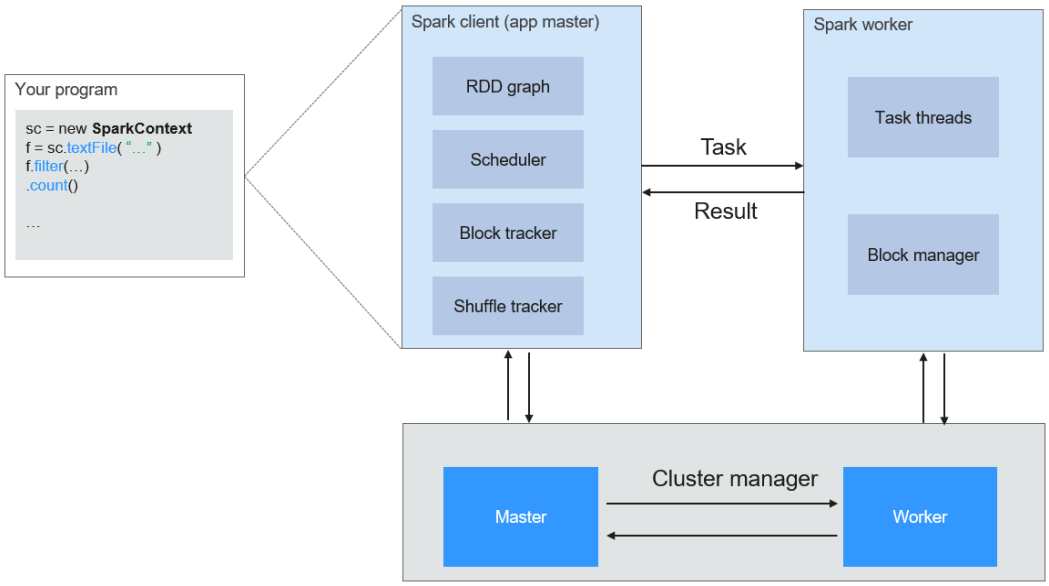
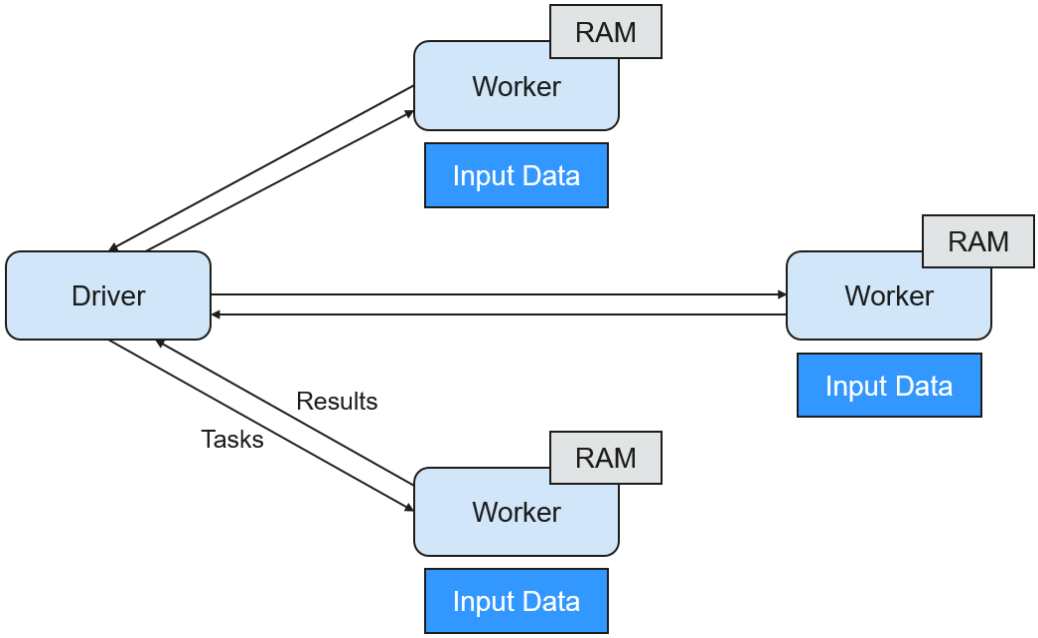


Figure 1-88 shows the Master and Worker modes adopted by Spark. A user submits an application on the Spark client, and then the scheduler divides a job into multiple tasks and sends the tasks to each Worker for execution. Each Worker reports the computation results to Driver (Master), and then the Driver aggregates and returns the results to the client.

Figure 1-88 Spark Master-Worker mode



Note the following about the architecture:

- Applications are isolated from each other.

  Each application has an independent executor process, and each executor starts multiple threads to execute tasks in parallel. Whether in terms of scheduling or task running on executors. Each driver independently schedules its own tasks. Different application tasks run on different JVMs, that is, different executors.
- Different Spark applications do not share data, unless data is stored in the external storage system such as HDFS.
- You are advised to deploy the Driver program in a location that is close to the Worker node because the Driver program schedules tasks in the cluster. For example, deploy the Driver program on the network where the Worker node is located.

Spark on YARN can be deployed in two modes:

- In Yarn-cluster mode, the Spark driver runs inside an ApplicationMaster process which is managed by Yarn in the cluster. After the ApplicationMaster is started, the client can exit without interrupting service running.
- In Yarn-client mode, the driver is started in the client process, and the ApplicationMaster process is used only to apply for resources from the Yarn cluster.

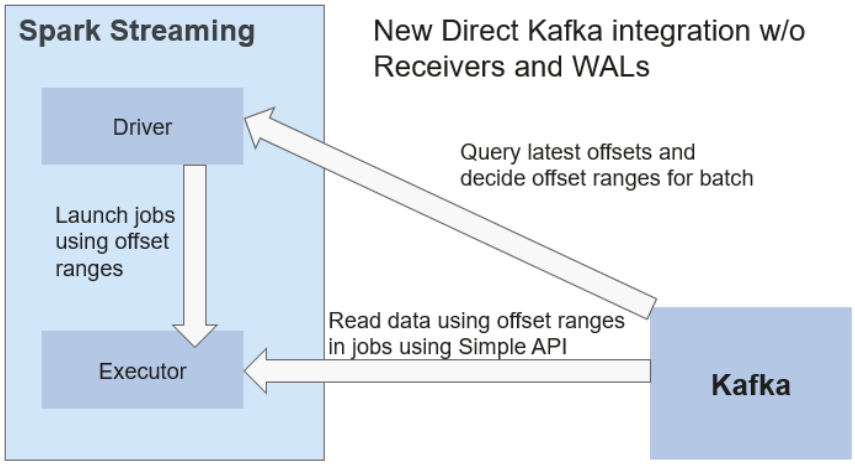
Spark Streaming Principle

Spark Streaming is a real-time computing framework built on the Spark, which expands the capability for processing massive streaming data. Currently, Spark supports the following data processing methods:

- Direct Streaming

In Direct Streaming approach, Direct API is used to process data. Take Kafka Direct API as an example. Direct API provides offset location that each batch range will read from, which is much simpler than starting a receiver to continuously receive data from Kafka and written data to write-ahead logs (WALs). Then, each batch job is running and the corresponding offset data is ready in Kafka. These offset information can be securely stored in the checkpoint file and read by applications that failed to start.

Figure 1-89 Data transmission through Direct Kafka API



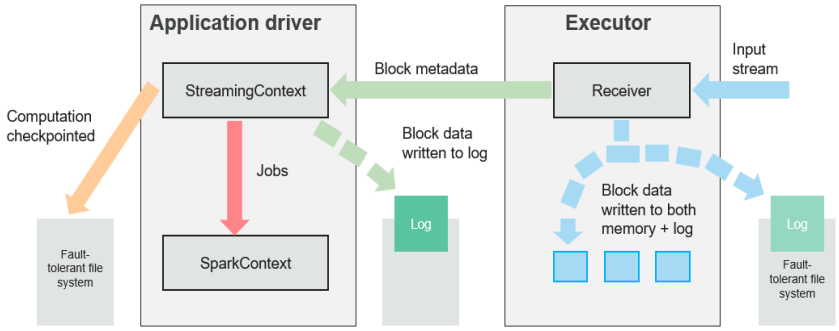
After the failure, Spark Streaming can read data from Kafka again and process the data segment. The processing result is the same no matter Spark Streaming fails or not, because the semantic is processed only once.

Direct API does not need to use the WAL and Receivers, and ensures that each Kafka record is received only once, which is more efficient. In this way, the Spark Streaming and Kafka can be well integrated, making streaming channels be featured with high fault-tolerance, high efficiency, and ease-of-use. Therefore, you are advised to use Direct Streaming to process data.

- Receiver

  When a Spark Streaming application starts (that is, when the driver starts), the related StreamingContext (the basis of all streaming functions) uses SparkContext to start the receiver to become a long-term running task. These receivers receive and save streaming data to the Spark memory for processing. **Figure 1-90** shows the data transfer lifecycle.

Figure 1-90 Data transfer lifecycle



a. Receive data (blue arrow).

   Receiver divides a data stream into a series of blocks and stores them in the executor memory. In addition, after WAL is enabled, it writes data to the WAL of the fault-tolerant file system.

b. Notify the driver (green arrow).

   The metadata in the received block is sent to StreamingContext in the driver. The metadata includes:

   - Block reference ID used to locate the data position in the Executor memory.
   - Block data offset information in logs (if the WAL function is enabled).

c. Process data (red arrow).

   For each batch of data, StreamingContext uses block information to generate resilient distributed datasets (RDDs) and jobs. StreamingContext executes jobs by running tasks to process blocks in the executor memory.

d. Periodically set checkpoints (orange arrows).

   For fault tolerance, StreamingContext periodically sets checkpoints and saves them to external file systems.

Fault Tolerance

Spark and its RDD allow seamless processing of failures of any Worker node in the cluster. Spark Streaming is built on top of Spark. Therefore, the Worker node of Spark Streaming also has the same fault tolerance capability. However, Spark Streaming needs to run properly in case of long-time running. Therefore, Spark must be able to recover from faults through the driver process (main process that coordinates all Workers). This poses challenges to the Spark driver fault-tolerance because the Spark driver may be any user application implemented in any computation mode. However, Spark Streaming has internal computation architecture. That is, it periodically executes the same Spark computation in each batch data. Such architecture allows it to periodically store checkpoints to reliable storage space and recover them upon the restart of Driver.

For source data such as files, the Driver recovery mechanism can ensure zero data loss because all data is stored in a fault-tolerant file system such as HDFS. However, for other data sources such as Kafka and Flume, some received data is cached only in memory and may be lost before being processed. This is caused by the distribution operation mode of Spark applications. When the driver process fails, all executors running in the Cluster Manager, together with all data in the memory, are terminated. To avoid such data loss, the WAL function is added to Spark Streaming.

WAL is often used in databases and file systems to ensure persistence of any data operation. That is, first record an operation to a persistent log and perform this operation on data. If the operation fails, the system is recovered by reading the log and re-applying the preset operation. The following describes how to use WAL to ensure persistence of received data:

Receiver is used to receive data from data sources such as Kafka. As a long-time running task in Executor, Receiver receives data, and also confirms received data if supported by data sources. Received data is stored in the Executor memory, and Driver delivers a task to Executor for processing.

After WAL is enabled, all received data is stored to log files in the fault-tolerant file system. Therefore, the received data does not lose even if Spark Streaming fails. Besides, receiver checks correctness of received data only after the data is pre-written into logs. Data that is cached but not stored can be sent again by data sources after the driver restarts. These two mechanisms ensure zero data loss. That is, all data is recovered from logs or re-sent by data sources.

To enable the WAL function, perform the following operations:

- Set **streamingContext.checkpoint** to configure the checkpoint directory, which is an HDFS file path used to store streaming checkpoints and WALs.
- Set **spark.streaming.receiver.writeAheadLog.enable** of SparkConf to **true** (the default value is **false**).

After WAL is enabled, all receivers have the advantage of recovering from reliable received data. You are advised to disable the multi-replica mechanism because the fault-tolerant file system of WAL may also replicate the data.

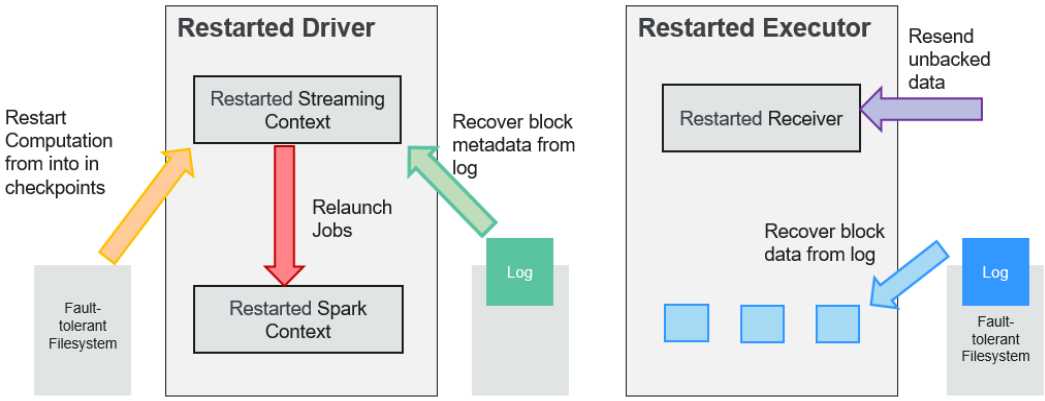
NOTE

The data receiving throughput is lowered after WAL is enabled. All data is written into the fault-tolerant file system. As a result, the write throughput of the file system and the network bandwidth for data replication may become the potential bottleneck. To solve this problem, you are advised to create more receivers to increase the degree of data receiving parallelism or use better hardware to improve the throughput of the fault-tolerant file system.

Recovery Process

When a failed driver is restarted, restart it as follows:

Figure 1-91 Computing recovery process



1. Recover computing. (Orange arrow)

   Use checkpoint information to restart Driver, reconstruct SparkContext and restart Receiver.

2. Recover metadata block. (Green arrow)

   This operation ensures that all necessary metadata blocks are recovered to continue the subsequent computing recovery.

3. Relaunch unfinished jobs. (Red arrow)

   Recovered metadata is used to generate RDDs and corresponding jobs for interrupted batch processing due to failures.

4. Read block data saved in logs. (Blue arrow)

   Block data is directly read from WALs during execution of the preceding jobs, and therefore all essential data reliably stored in logs is recovered.

5. Resend unconfirmed data. (Purple arrow)

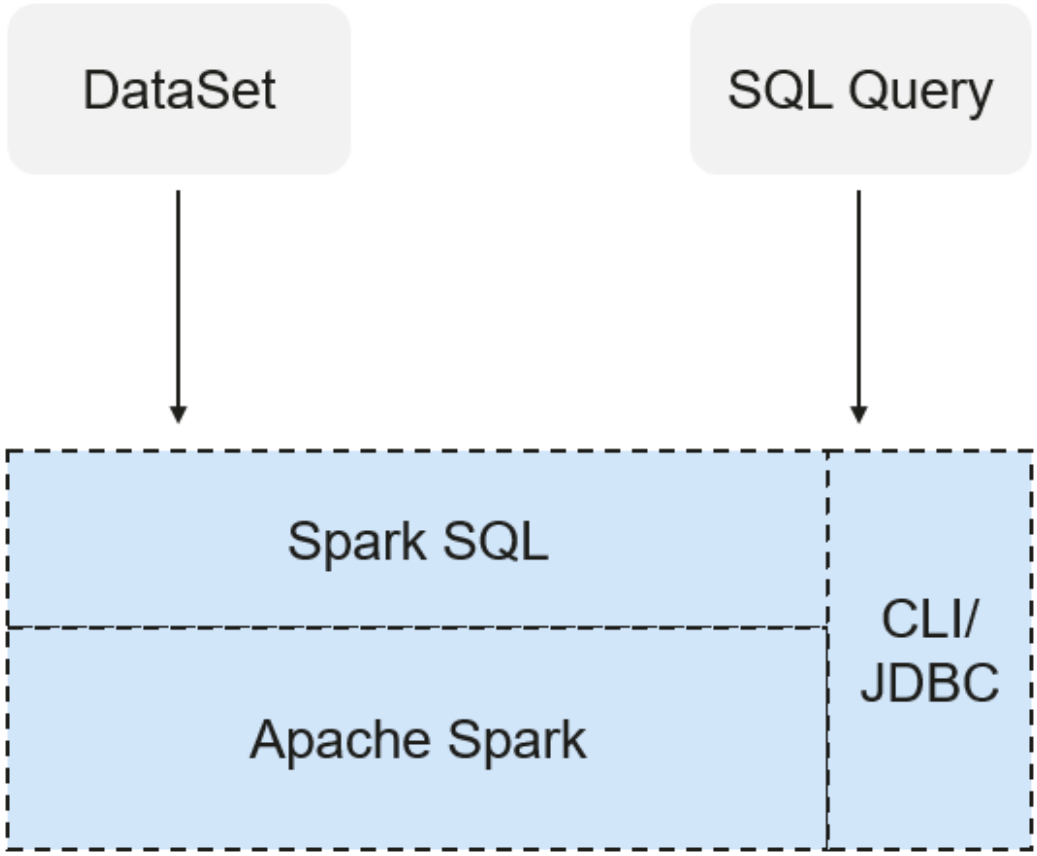
   Data that is cached but not stored to logs upon failures is re-sent by data sources, because the receiver does not confirm the data.

Therefore, by using WALs and reliable Receiver, Spark Streaming can avoid input data loss caused by Driver failures.

SparkSQL and DataSet Principle

SparkSQL

Figure 1-92 SparkSQL and DataSet



Spark SQL is a module for processing structured data. In Spark application, SQL statements or DataSet APIs can be seamlessly used for querying structured data.

Spark SQL and DataSet also provide a universal method for accessing multiple data sources such as Hive, CSV, Parquet, ORC, JSON, and JDBC. These data sources also allow data interaction. Spark SQL reuses the Hive frontend processing logic and metadata processing module. With the Spark SQL, you can directly query existing Hive data.

In addition, Spark SQL also provides API, CLI, and JDBC APIs, allowing diverse accesses to the client.

Spark SQL Native DDL/DML

In Spark 1.5, lots of Data Definition Language (DDL)/Data Manipulation Language (DML) commands are pushed down to and run on the Hive, causing coupling with the Hive and inflexibility such as unexpected error reports and results.

Spark 3.1.1 realizes command localization and replaces the Hive with Spark SQL Native DDL/DML to run DDL/DML commands. Additionally, the decoupling from the Hive is realized and commands can be customized.

DataSet

A DataSet is a strongly typed collection of domain-specific objects that can be transformed in parallel using functional or relational operations. Each Dataset also has an untyped view called a DataFrame, which is a Dataset of Row.

The DataFrame is a structured and distributed dataset consisting of multiple columns. The DataFrame is equal to a table in the relationship database or the DataFrame in the R/Python. The DataFrame is the most basic concept in the Spark SQL, which can be created by using multiple methods, such as the structured dataset, Hive table, external database or RDD.

Operations available on DataSets are divided into transformations and actions.

- A transformation operation can generate a new DataSet,
  for example, **map**, **filter**, **select**, and **aggregate (groupBy)**.
- An action operation can trigger computation and return results,
  for example, **count**, **show**, or write data to the file system.

You can use either of the following methods to create a DataSet:

- The most common way is by pointing Spark to some files on storage systems, using the **read** function available on a SparkSession.
  ```
  val people = spark.read.parquet("...").as[Person]  // Scala
  DataSet<Person> people = spark.read().parquet("...").as(Encoders.bean(Person.class));//Java
  ```
- You can also create a DataSet using the transformation operation available on an existing one.
  For example, apply the map operation on an existing DataSet to create a DataSet:
  ```
  val names = people.map(_.name) // In Scala: names is Dataset.
  Dataset<String> names = people.map((Person p) -> p.name, Encoders.STRING)); // Java
  ```

CLI and JDBCServer

In addition to programming APIs, Spark SQL also provides the CLI/JDBC APIs.

- Both **spark-shell** and **spark-sql** scripts can provide the CLI for debugging.
- JDBCServer provides JDBC APIs. External systems can directly send JDBC requests to calculate and parse structured data.

SparkSession Principle

SparkSession is a unified API for Spark programming and can be regarded as a unified entry for reading data. SparkSession provides a single entry point to perform many operations that were previously scattered across multiple classes, and also provides accessor methods to these older classes to maximize compatibility.

A SparkSession can be created using a builder pattern. The builder will automatically reuse the existing SparkSession if there is a SparkSession; or create a SparkSession if it does not exist. During I/O transactions, the configuration item settings in the builder are automatically synchronized to Spark and Hadoop.

```
import org.apache.spark.sql.SparkSession
val sparkSession = SparkSession.builder
  .master("local")
  .appName("my-spark-app")
  .config("spark.some.config.option", "config-value")
  .getOrCreate()
```

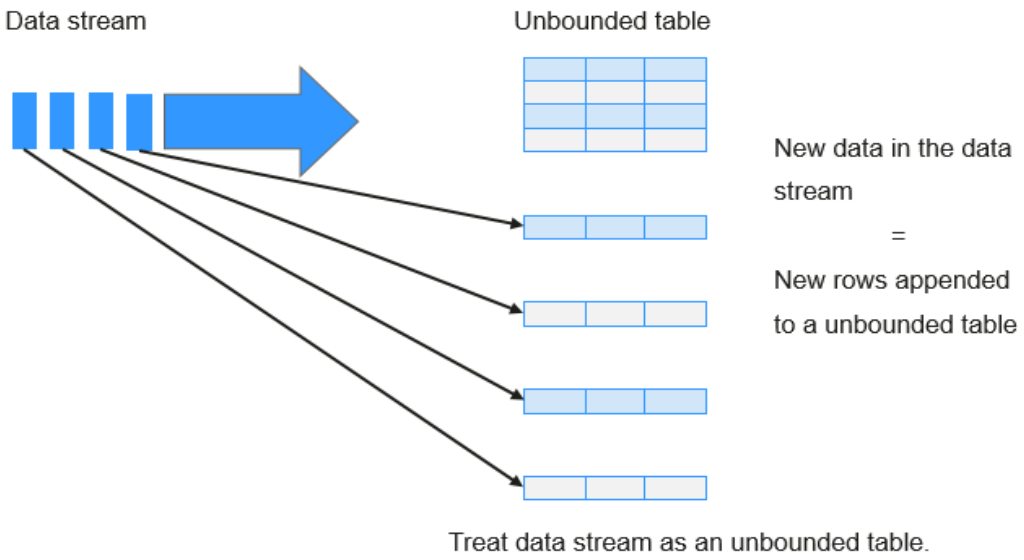
- SparkSession can be used to execute SQL queries on data and return results as DataFrame.
  ```
  sparkSession.sql("select * from person").show
  ```
- SparkSession can be used to set configuration items during running. These configuration items can be replaced with variables in SQL statements.
  ```
  sparkSession.conf.set("spark.some.config", "abcd")
  sparkSession.conf.get("spark.some.config")
  sparkSession.sql("select ${spark.some.config}")
  ```
- SparkSession also includes a "catalog" method that contains methods to work with Metastore (data catalog). After this method is used, a dataset is returned, which can be run using the same Dataset API.
  ```
  val tables = sparkSession.catalog.listTables()
  val columns = sparkSession.catalog.listColumns("myTable")
  ```
- Underlying SparkContext can be accessed by SparkContext API of SparkSession.
  ```
  val sparkContext = sparkSession.sparkContext
  ```

Structured Streaming Principle

Structured Streaming is a stream processing engine built on the Spark SQL engine. You can use the Dataset/DataFrame API in Scala, Java, Python, or R to express streaming aggregations, event-time windows, and stream-stream joins. If streaming data is incrementally and continuously produced, Spark SQL will continue to process the data and synchronize the result to the result set. In addition, the system ensures end-to-end exactly-once fault-tolerance guarantees through checkpoints and WALs.

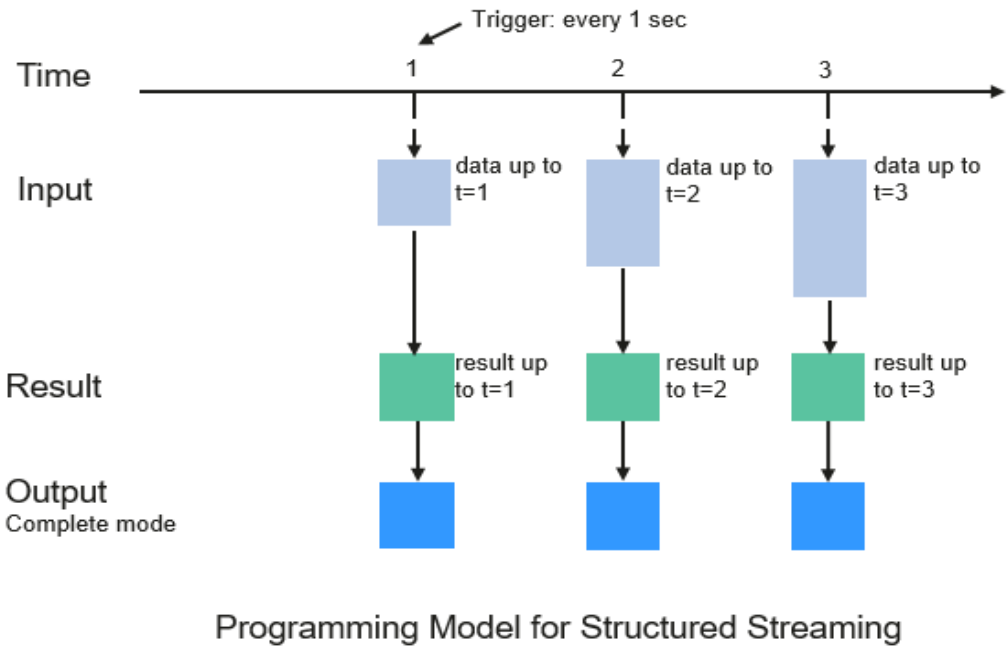
The core of Structured Streaming is to take streaming data as an incremental database table. Similar to the data block processing model, the streaming data processing model applies query operations on a static database table to streaming computing, and Spark uses standard SQL statements for query, to obtain data from the incremental and unbounded table.

Figure 1-93 Unbounded table of Structured Streaming



Each query operation will generate a result table. At each trigger interval, updated data will be synchronized to the result table. Whenever the result table is updated, the updated result will be written into an external storage system.

Figure 1-94 Structured Streaming data processing model



Storage modes of Structured Streaming at the output phase are as follows:

- Complete Mode: The updated result sets are written into the external storage system. The write operation is performed by a connector of the external storage system.

- Append Mode: If an interval is triggered, only added data in the result table will be written into an external system. This is applicable only on the queries where existing rows in the result table are not expected to change.
- Update Mode: If an interval is triggered, only updated data in the result table will be written into an external system, which is the difference between the Complete Mode and Update Mode.

Basic Concepts

- **RDD**

  Resilient Distributed Dataset (RDD) is a core concept of Spark. It indicates a read-only and partitioned distributed dataset. Partial or all data of this dataset can be cached in the memory and reused between computations.

  RDD Creation

  - An RDD can be created from the input of HDFS or other storage systems that are compatible with Hadoop.
  - A new RDD can be converted from a parent RDD.
  - An RDD can be converted from a collection of datasets through encoding.

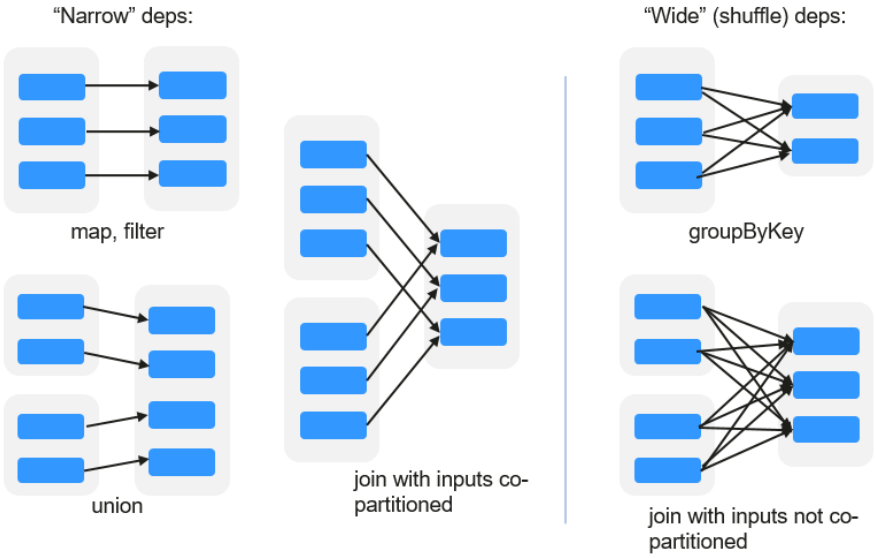
  RDD Storage

  - You can select different storage levels to store an RDD for reuse. (There are 11 storage levels to store an RDD.)
  - By default, the RDD is stored in the memory. When the memory is insufficient, the RDD overflows to the disk.

- **RDD Dependency**

  The RDD dependency includes the narrow dependency and wide dependency.

  Figure 1-95 RDD dependency

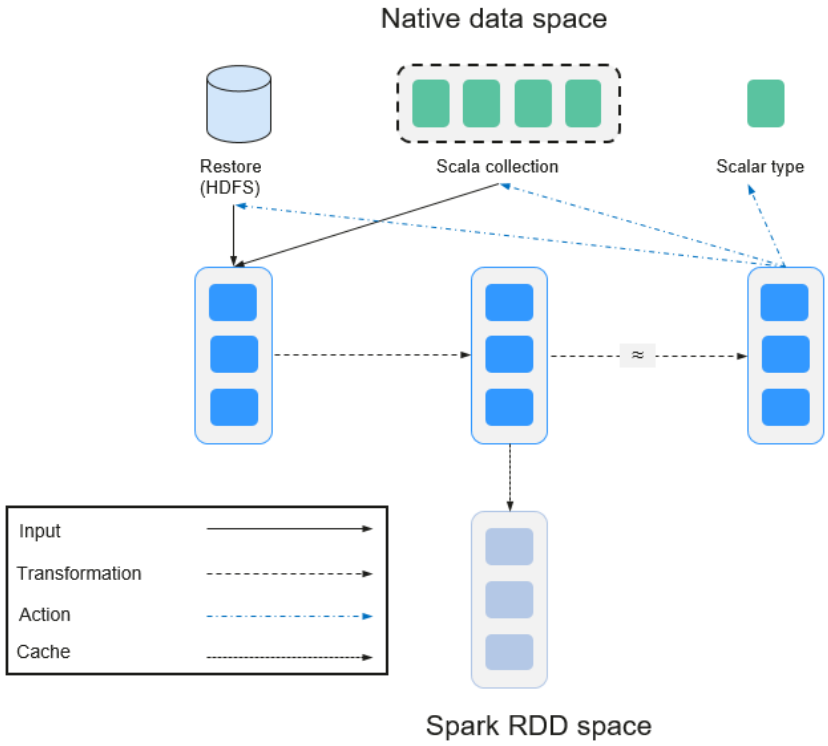
  - **Narrow dependency**: Each partition of the parent RDD is used by at most one partition of the child RDD.
  - **Wide dependency**: Partitions of the child RDD depend on all partitions of the parent RDD.

The narrow dependency facilitates the optimization. Logically, each RDD operator is a fork/join (the join is not the join operator mentioned above but the barrier used to synchronize multiple concurrent tasks); fork the RDD to each partition, and then perform the computation. After the computation, join the results, and then perform the fork/join operation on the next RDD operator. It is uneconomical to directly translate the RDD into physical implementation. The first is that every RDD (even intermediate result) needs to be physicalized into memory or storage, which is time-consuming and occupies much space. The second is that as a global barrier, the join operation is very expensive and the entire join process will be slowed down by the slowest node. If the partitions of the child RDD narrowly depend on that of the parent RDD, the two fork/join processes can be combined to implement classic fusion optimization. If the relationship in the continuous operator sequence is narrow dependency, multiple fork/join processes can be combined to reduce a large number of global barriers and eliminate the physicalization of many RDD intermediate results, which greatly improves the performance. This is called pipeline optimization in Spark.

- **Transformation and Action (RDD Operations)**

  Operations on RDD include transformation (the return value is an RDD) and action (the return value is not an RDD). **Figure 1-96** shows the RDD operation process. The transformation is lazy, which indicates that the transformation from one RDD to another RDD is not immediately executed. Spark only records the transformation but does not execute it immediately. The real computation is started only when the action is started. The action returns results or writes the RDD data into the storage system. The action is the driving force for Spark to start the computation.

Figure 1-96 RDD operation



The data and operation model of RDD are quite different from those of Scala.

```
val file = sc.textFile("hdfs://...")
val errors = file.filter(_.contains("ERROR"))
errors.cache()
errors.count()
```

a. The textFile operator reads log files from the HDFS and returns files (as an RDD).

b. The filter operator filters rows with **ERROR** and assigns them to errors (a new RDD). The filter operator is a transformation.

c. The cache operator caches errors for future use.

d. The count operator returns the number of rows of errors. The count operator is an action.

Transformation includes the following types:

– The RDD elements are regarded as simple elements.

  The input and output has the one-to-one relationship, and the partition structure of the result RDD remains unchanged, for example, map.

  The input and output has the one-to-many relationship, and the partition structure of the result RDD remains unchanged, for example, flatMap (one element becomes a sequence containing multiple elements after map and then flattens to multiple elements).

  The input and output has the one-to-one relationship, but the partition structure of the result RDD changes, for example, union (two RDDs integrates to one RDD, and the number of partitions becomes the sum of the number of partitions of two RDDs) and coalesce (partitions are reduced).

  Operators of some elements are selected from the input, such as filter, distinct (duplicate elements are deleted), subtract (elements only exist in this RDD are retained), and sample (samples are taken).

– The RDD elements are regarded as key-value pairs.

  Perform the one-to-one calculation on the single RDD, such as mapValues (the partition mode of the source RDD is retained, which is different from map).

  Sort the single RDD, such as sort and partitionBy (partitioning with consistency, which is important to the local optimization).

  Restructure and reduce the single RDD based on key, such as groupByKey and reduceByKey.

  Join and restructure two RDDs based on the key, such as join and cogroup.

  > NOTE
  >
  > The later three operations involving sorting are called shuffle operations.

Action includes the following types:

– Generate scalar configuration items, such as **count** (the number of elements in the returned RDD), **reduce**, **fold/aggregate** (the number of scalar configuration items that are returned), and **take** (the number of elements before the return).

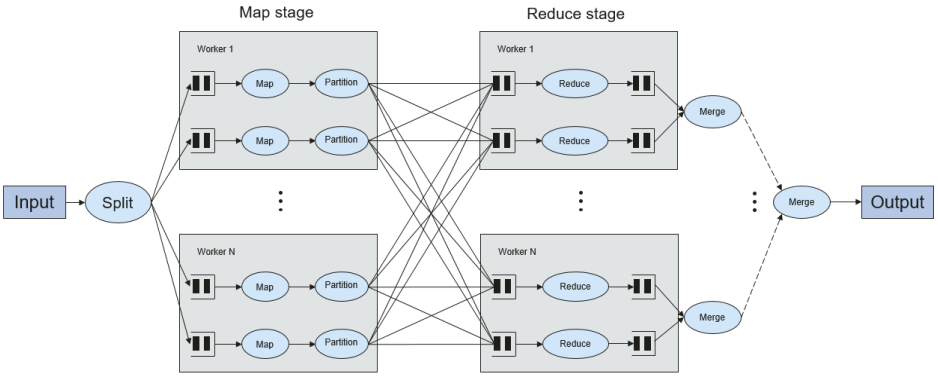
  - Generate the Scala collection, such as **collect** (import all elements in the RDD to the Scala collection) and **lookup** (look up all values corresponds to the key).
  - Write data to the storage, such as **saveAsTextFile** (which corresponds to the preceding **textFile**).
  - Check points, such as the **checkpoint** operator. When Lineage is quite long (which occurs frequently in graphics computation), it takes a long period of time to execute the whole sequence again when a fault occurs. In this case, checkpoint is used as the check point to write the current data to stable storage.

- **Shuffle**

  Shuffle is a specific phase in the MapReduce framework, which is located between the Map phase and the Reduce phase. If the output results of Map are to be used by Reduce, the output results must be hashed based on a key and distributed to each Reducer. This process is called Shuffle. Shuffle involves the read and write of the disk and the transmission of the network, so that the performance of Shuffle directly affects the operation efficiency of the entire program.

  The figure below shows the entire process of the MapReduce algorithm.

  Figure 1-97 Algorithm process



  Shuffle is a bridge to connect data. The following describes the implementation of shuffle in Spark.

  Shuffle divides a job of Spark into multiple stages. The former stages contain one or more ShuffleMapTasks, and the last stage contains one or more ResultTasks.

- **Spark Application Structure**

  The Spark application structure includes the initialized SparkContext and the main program.

  - Initialized SparkContext: constructs the operating environment of the Spark Application.

    Constructs the SparkContext object. The following is an example:

    ```
    new SparkContext(master, appName, [SparkHome], [jars])
    ```

    Parameter description:

    master: indicates the link string. The link modes include local, Yarn-cluster, and Yarn-client.

appName: indicates the application name.

SparkHome: indicates the directory where Spark is installed in the cluster.

jars: indicates the code and dependency package of an application.

– Main program: processes data.

- **Spark Shell Commands**

  The basic Spark shell commands support the submission of Spark applications. The Spark shell commands are as follows:

```
./bin/spark-submit \
  --class <main-class> \
  --master <master-url> \
  ... # other options
  <application-jar> \
  [application-arguments]
```

  Parameter description:

  --class: indicates the name of the class of a Spark application.

  --master: indicates the master to which the Spark application links, such as Yarn-client and Yarn-cluster.

  application-jar: indicates the path of the JAR file of the Spark application.

  application-arguments: indicates the parameter required to submit the Spark application. This parameter can be left blank.

- **Spark JobHistory Server**

  The Spark web UI is used to monitor the details in each phase of the Spark framework of a running or historical Spark job and provide the log display, which helps users to develop, configure, and optimize the job in more fine-grained units.

1.4.23.2 Spark HA Solution

Spark Multi-Active Instance HA Principles and Implementation Solution

Based on existing JDBCServer in the community, multi-active-instance mode is used to achieve HA. In this mode, multiple JDBCServers coexist in the cluster and the client can randomly connect any JDBCServer to perform service operations. When one or multiple JDBCServers stop working, a client can connect to another normal JDBCServer.

Compared with active/standby HA mode, multi-active instance mode has following advantages:

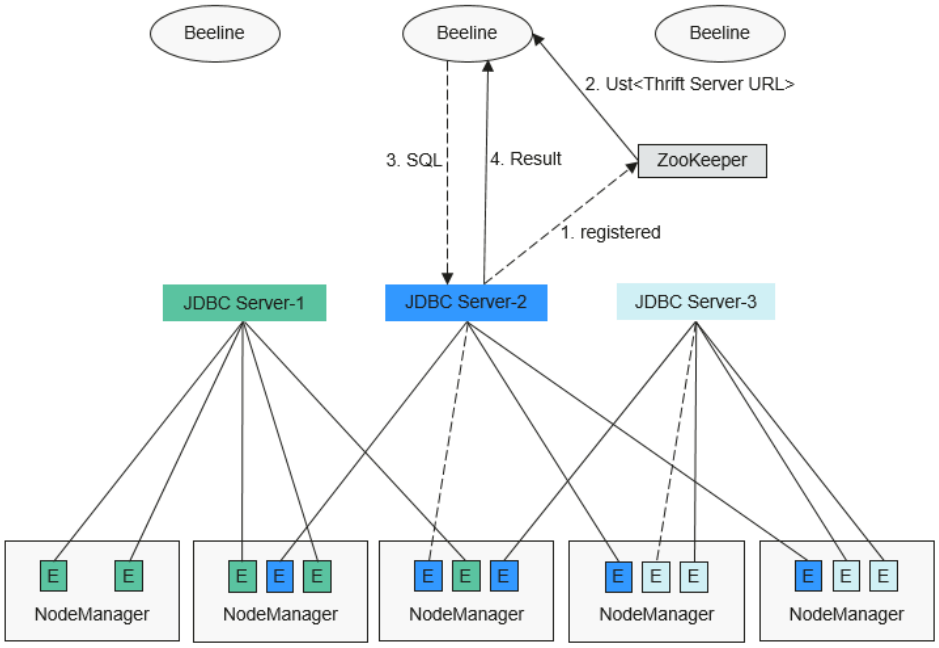
- In active/standby HA, when the active/standby switchover occurs, the unavailable period cannot be controlled by JDBCServer, but it depends on Yarn service resources.
- In Spark, the Thrift JDBC similar to HiveServer2 provides services and users access services through Beeline and JDBC API. Therefore, the processing capability of the JDBCServer cluster depends on the single-point capability of the primary server, and the scalability is insufficient.

The multi-active instance HA mode not only can prevent service interruption caused by switchover, but also enables cluster scale-out to improve high concurrency.

- **Implementation**

  The following figure shows the basic principle of multi-active instance HA of Spark JDBCServer.

  Figure 1-98 Spark JDBCServer HA



1. When a JDBCServer is started, it registers with ZooKeeper by writing node information in a specified directory. Node information includes the instance IP address, port number, version, and serial number.
2. To connect to JDBCServer, the client must specify the namespace, which is the directory of JDBCServer instances in ZooKeeper. During the connection, a JDBCServer instance is randomly selected from the specified namespace.
3. After the connection succeeds, the client sends SQL statements to JDBCServer.
4. JDBCServer executes received SQL statements and returns results to the client.

If multi-active instance HA of Spark JDBCServer is enabled, all JDBCServer instances are independent and equivalent. When one JDBCServer instance is interrupted during upgrade, other JDBCServer instances can accept the connection request from the client.

The rules below must be followed in the multi-active instance HA of Spark JDBCServer.

- If a JDBCServer instance exits abnormally, no other instance will take over the sessions and services running on the abnormal instance.
- When the JDBCServer process is stopped, corresponding nodes are deleted from ZooKeeper.
- The client randomly selects the server, which may result in uneven session allocation caused by random distribution of policy results, and finally result in load imbalance of instances.
- After the instance enters the maintenance mode (in which no new connection requests from clients are accepted), services running on the instance may fail when the decommissioning times out.

- **URL Connection**
  - Multi-active instance mode

    In multi-active instance mode, the client reads content from the ZooKeeper node and connects to JDBCServer. The connection strings are list below.

    - Security mode:

      If Kinit authentication is enabled, the JDBCURL is as follows:

      jdbc:hive2://<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.**<*System domain name*>**@**<*System domain name*>**;**

      NOTE

      - In the above JDBCURL, **<zkNode_IP>:<zkNode_Port>** indicates the ZooKeeper URL. Use commas (,) to separate multiple URLs,

        Example: 192.168.81.37:2181,192.168.195.232:2181,192.168.169.84:2181.
      - **sparkthriftserver2x** indicates the ZooKeeper directory, where a random JDBCServer instance is connected to the client.

      For example, when you use Beeline client to connect JDBCServer, run the following command:

      sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://**<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.**<*System domain name*>**@**<*System domain name*>**;"**

      If Keytab authentication is enabled, the JDBCURL is as follows:

      jdbc:hive2://<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.**<*System domain name*>@<*System domain name*>**;user.principal=**<*principal_name*>**;user.keytab=**<*path_to_keytab*>

      In the above URL, <*principal_name*> indicates the principal of the Kerberos user, for example, **test@**<*System domain name*>; <*path_to_keytab*> indicates the Keytab file path corresponding to <*principal_name*>, for example, **/opt/auth/test/user.keytab**.

    - Common mode:

      jdbc:hive2://<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;**

      For example, when you use Beeline client, in normal mode, for connection, run the following command:

      sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://**<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;"**

  - Non-multi-active instance mode

    In this mode, a client connects to a specified JDBCServer node. Compared with multi-active instance mode, the connection string in this mode does

not contain **serviceDiscoveryMode** and **zooKeeperNamespace** parameters about ZooKeeper.

For example, when you use Beeline client, in security mode, to connect JDBCServer in non-multi-active instance mode, run the following command:

sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://***<server_IP>:<server_Port>***/;user.principal=spark/hadoop.***<System domain name>***@***<System domain name>***;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.***<System domain name>***@***<System domain name>***;"**

NOTE

- In the above command, **<server_IP>:<server_Port>** indicates the URL of the specified JDBCServer node.
- **CLIENT_HOME** indicates the client path.

Except the connection method, other operations of JDBCServer API in the two modes are the same. Spark JDBCServer is another implementation of HiveServer2 in Hive.

Spark Multi-Tenant HA

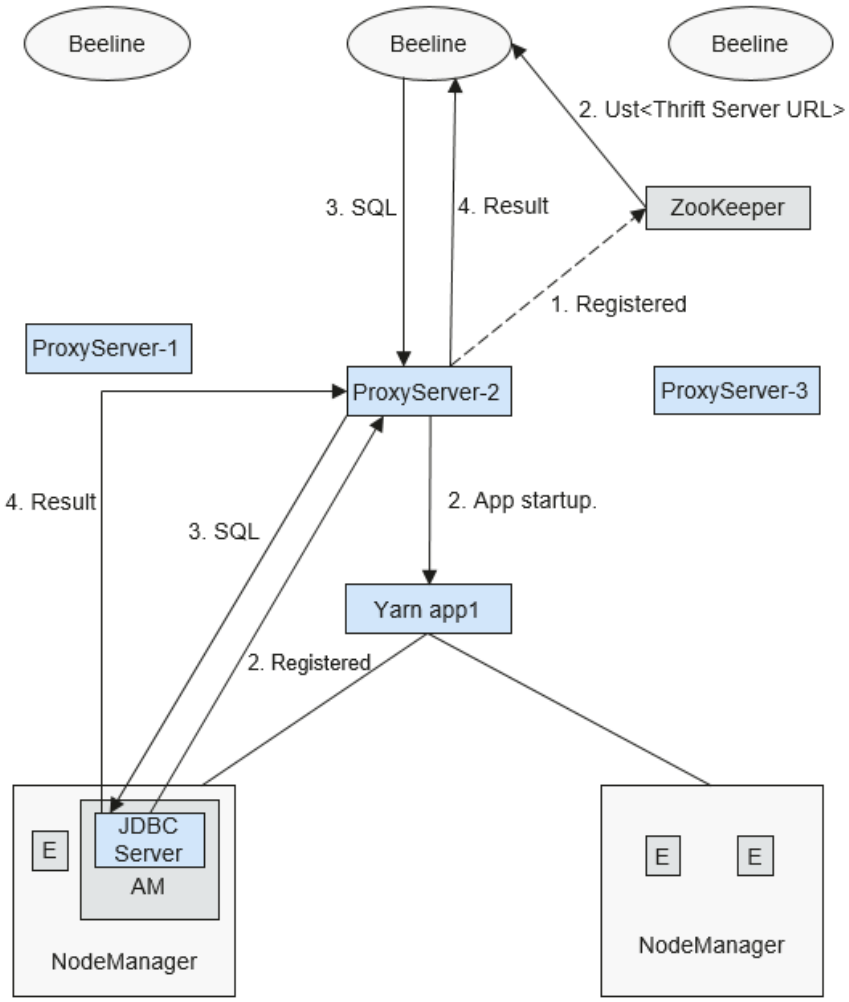
In the JDBCServer multi-active instance solution, JDBCServer uses the Yarn-client mode, but there is only one Yarn resource queue available. To solve this resource limitation problem, the multi-tenant mode is introduced.

In multi-tenant mode, JDBCServers are bound with tenants. Each tenant corresponds to one or more JDBCServers, and a JDBCServer provides services for only one tenant. Different tenants can be configured with different Yarn queues to implement resource isolation. In addition, JDBCServer can be dynamically started as required to avoid resource waste.

- **Implementation**

  Figure 1-99 shows the HA solution of the multi-tenant mode.

Figure 1-99 Multi-tenant mode of Spark JDBCServer



a. When ProxyServer is started, it registers with ZooKeeper by writing node information in a specified directory. Node information includes the instance IP address, port number, version, and serial number.

   NOTE

   In multi-tenant mode, the JDBCServer instance refers to the ProxyServer (JDBCServer proxy).

b. To connect to ProxyServer, the client must specify a namespace, which is the directory of the ProxyServer instance where you want to access ZooKeeper. When the client connects to the ProxyServer, a random instance under the namespace is selected for connection. For details about the URL, see **URL Connection Overview**.

c. After the client successfully connects to the ProxyServer, which first checks whether the JDBCServer of a tenant exists. If yes, Beeline connects the JDBCServer. If no, a new JDBCServer is started in Yarn-cluster mode. After the startup of JDBCServer, ProxyServer obtains the IP address of the JDBCServer and establishes the connection between Beeline and JDBCServer.

d. The client sends SQL statements to ProxyServer, which forwards statements to the connected JDBCServer. JDBCServer returns the results to ProxyServer, which then returns the results to the client.

In the multi-active instance HA mode, all instances are independent and equivalent. If one instance is interrupted during upgrade, other instances can accept the connection request from the client.

- **URL Connection Overview**
  - Multi-tenant mode

    In multi-tenant mode, the client reads content from the ZooKeeper node and connects to ProxyServer. The connection strings are list below.

    - Security mode:

      If Kinit authentication is enabled, the client URL is as follows:

      jdbc:hive2://<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.**<*System domain name*>@<*System domain name*>;

      NOTE

      - In the above URL, **<zkNode_IP>:<zkNode_Port>** indicates the ZooKeeper URL. Use commas (,) to separate multiple URLs,

        Example:
        192.168.81.37:2181,192.168.195.232:2181,192.168.169.84:2181.
      - **sparkthriftserver2x** indicates the ZooKeeper directory, where a random JDBCServer instance is connected to the client.

      For example, when you use Beeline client for connection, run the following command:

      sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://**<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.**<*System domain name*>**@**<*System domain name*>**;"**

      If Keytab authentication is enabled, the URL is as follows:

      jdbc:hive2://<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.**<*System domain name*>@<*System domain name*>**;user.principal=**<*principal_name*>**;user.keytab=**<*path_to_keytab*>

      In the above URL, <*principal_name*> indicates the principal of the Kerberos user, for example, **test@**<*System domain name*>; <*path_to_keytab*> indicates the Keytab file path corresponding to <*principal_name*>, for example, **/opt/auth/test/user.keytab**.

    - Common mode:

      jdbc:hive2://<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=**sparkthriftserver2x**;

      For example, run the following command when you use Beeline client for connection in normal mode:

      sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://**<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkN*

ode3_IP>:<zkNode3_Port>/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;"

- Non-multi-tenant mode

  In non-multi-tenant mode, a client connects to a specified JDBCServer node. Compared with multi-tenant instance mode, the connection string in this mode does not contain **serviceDiscoveryMode** and **zooKeeperNamespace** parameters about ZooKeeper.

  For example, when you use Beeline client to connect JDBCServer in non-multi-tenant instance mode, run the following command:

  sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://***<server_IP>:<server_Port>*/;user.principal=spark/hadoop.*<System domain name>@<System domain name>*;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.*<System domain name>@<System domain name>*;"

  NOTE

  - In the above command, **<server_IP>:<server_Port>** indicates the URL of the specified JDBCServer node.
  - **CLIENT_HOME** indicates the client path.

  Except the connection method, other operations of JDBCServer API in multi-tenant mode and non-multi-tenant mode are the same. Spark JDBCServer is another implementation of HiveServer2 in Hive.

  Specifying a Tenant

  Generally, the client submitted by a user connects to the default JDBCServer of the tenant to which the user belongs. If you want to connect the client to the JDBCServer of a specified tenant, add the **--hiveconf mapreduce.job.queuename** parameter.

  If you use Beeline client for connection, run the following command (**aaa** is the tenant name):

  beeline --hiveconf mapreduce.job.queuename=aaa -u 'jdbc:hive2://192.168.39.30:2181,192.168.40.210:2181,192.168.215.97:2181;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.*<System domain name>@<System domain name>*;'

1.4.23.3 Relationship Among Spark, HDFS, and Yarn

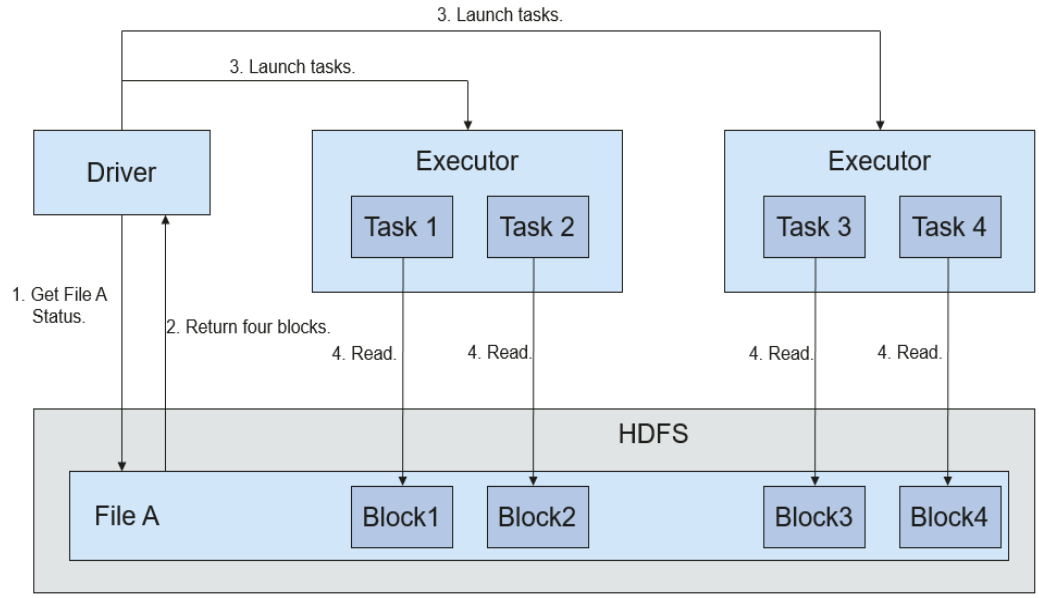
Relationship Between Spark and HDFS

Data computed by Spark comes from multiple data sources, such as local files and HDFS. Most data computed by Spark comes from the HDFS. The HDFS can read data in large scale for parallel computing. After being computed, data can be stored in the HDFS.

Spark involves Driver and Executor. Driver schedules tasks and Executor runs tasks.

Figure 1-100 shows the process of reading a file.

Figure 1-100 File reading process

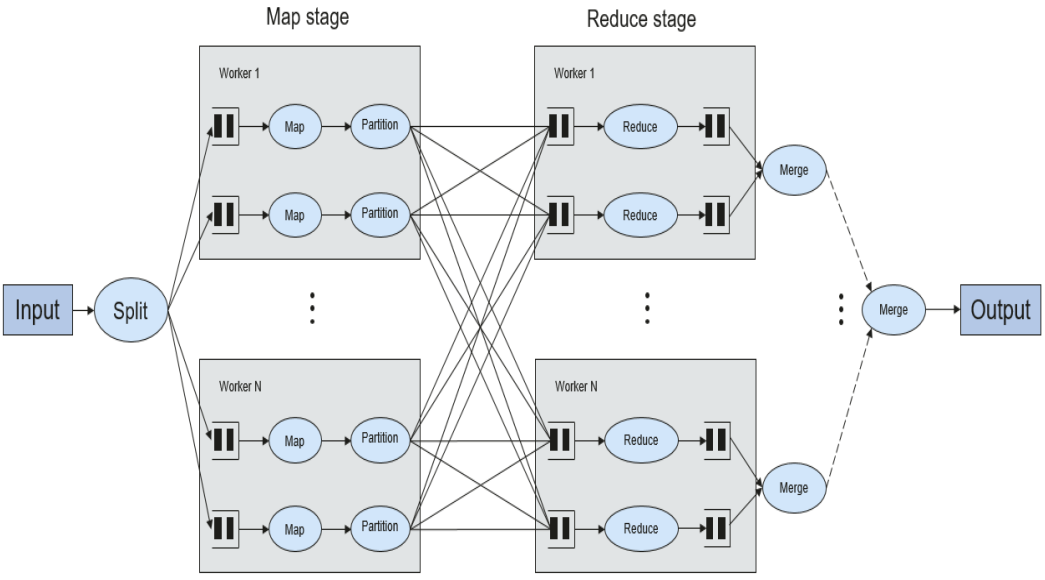


The file reading process is as follows:

1. Driver interconnects with the HDFS to obtain the information of File A.
2. The HDFS returns the detailed block information about this file.
3. Driver sets a parallel degree based on the block data amount, and creates multiple tasks to read the blocks of this file.
4. Executor runs the tasks and reads the detailed blocks as part of the Resilient Distributed Dataset (RDD).

Figure 1-101 shows the process of writing data to a file.

Figure 1-101 File writing process



The file writing process is as follows:

1. Driver creates a directory where the file is to be written.
2. Based on the RDD distribution status, the number of tasks related to data writing is computed, and these tasks are sent to Executor.
3. Executor runs these tasks, and writes the RDD data to the directory created in 1.

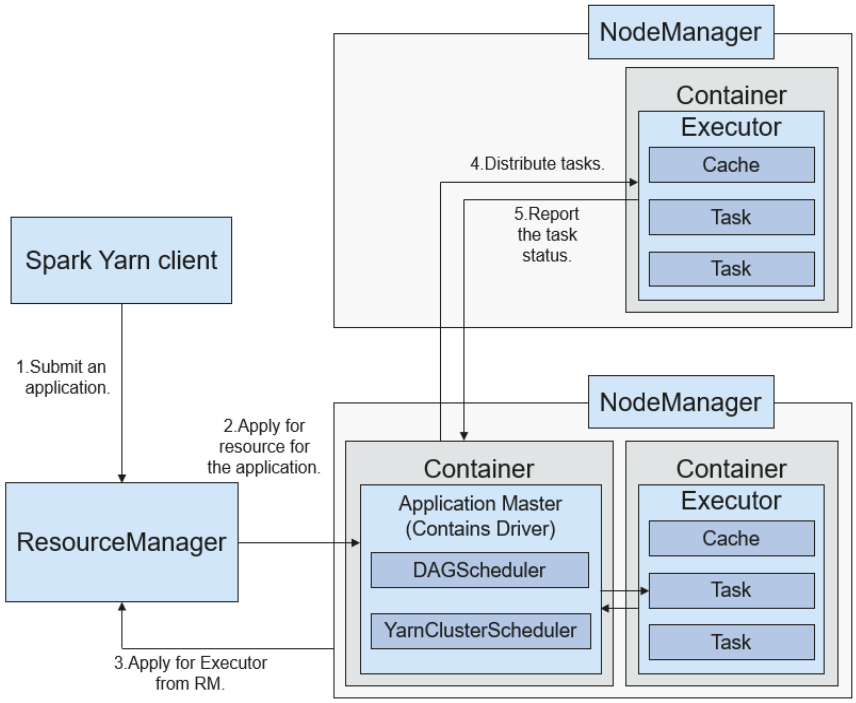
Relationship Between Spark and Yarn

The Spark computing and scheduling can be implemented using Yarn mode. Spark enjoys the computing resources provided by Yarn clusters and runs tasks in a distributed way. Spark on Yarn has two modes: Yarn-cluster and Yarn-client.

- Yarn-cluster mode

  Figure 1-102 shows the running framework of Spark on Yarn-cluster.

  Figure 1-102 Spark on Yarn-cluster operation framework



  Spark on Yarn-cluster implementation process:

  a. The client generates the application information, and then sends the information to ResourceManager.
  b. ResourceManager allocates the first container (ApplicationMaster) to SparkApplication and starts driver on the container.
  c. ApplicationMaster applies for resources from ResourceManager to run the container.

     ResourceManager allocates the container to ApplicationMaster, which communicates with NodeManager, and starts the executor in the obtained container. After the executor is started, it registers with the driver and applies for tasks.

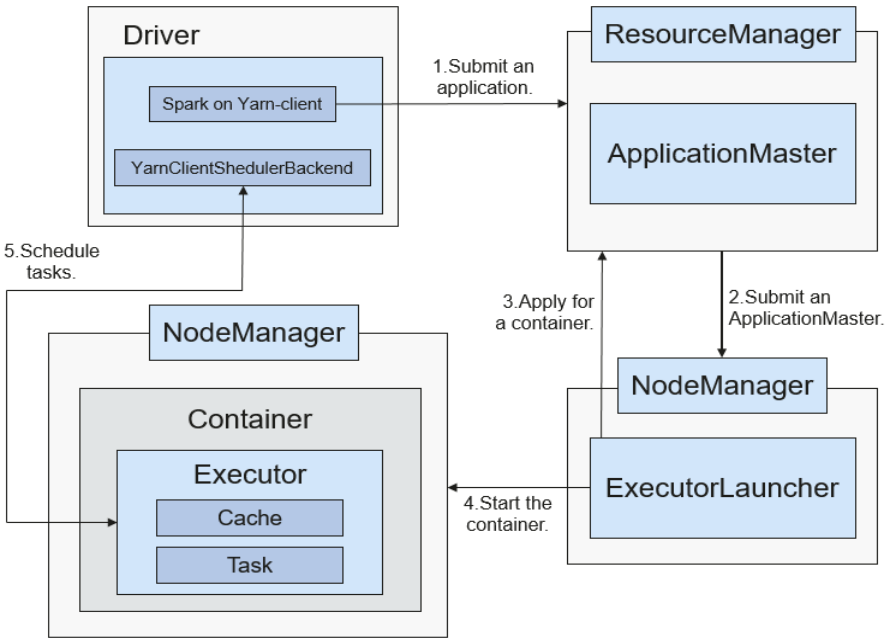
d. The driver allocates tasks to the executor.

e. The executor runs tasks and reports the operating status to the driver.

- Yarn-client mode

  Figure 1-103 shows the running framework of Spark on Yarn-cluster.

  Figure 1-103 Spark on Yarn-client operation framework



  Spark on Yarn-client implementation process:

  NOTE

  In Yarn-client mode, Driver is deployed on the client and started on the client. In Yarn-client mode, the client of the earlier version is incompatible. You are advised to use the Yarn-cluster mode.

  a. The client sends the Spark application request to ResourceManager, then ResourceManager returns the results. The results include information such as Application ID and the maximum and minimum available resources. The client packages all information required to start ApplicationMaster, and sends the information to ResourceManager.

  b. After receiving the request, ResourceManager finds a proper node for ApplicationMaster and starts it on this node. ApplicationMaster is a role in Yarn, and the process name in Spark is ExecutorLauncher.

  c. Based on the resource requirements of each task, ApplicationMaster can apply for a series of Containers to run tasks from ResourceManager.

  d. After receiving the newly allocated container list (from ResourceManager), ApplicationMaster sends information to the related NodeManagers to start the containers.

     ResourceManager allocates the containers to ApplicationMaster, which communicates with the related NodeManagers, and starts the executors in the obtained containers. After the executors are started, it registers with drivers and applies for tasks.

NOTE

Running containers are not suspended and resources are not released.

e. The drivers allocate tasks to the executors. The executor executes tasks and reports the operating status to the driver.

1.4.23.4 Spark Enhanced Open Source Feature: Optimized SQL Query of Cross-Source Data

Scenario

Enterprises usually store massive data, such as from various databases and warehouses, for management and information collection. However, diversified data sources, hybrid dataset structures, and scattered data storage lower query efficiency.

The open source Spark only supports simple filter pushdown during querying of multi-source data. The SQL engine performance is deteriorated due of a large amount of unnecessary data transmission. The pushdown function is enhanced, so that **aggregate**, complex **projection**, and complex **predicate** can be pushed to data sources, reducing unnecessary data transmission and improving query performance.

Only the JDBC data source supports pushdown of query operations, such as **aggregate**, **projection**, **predicate**, **aggregate over inner join**, and **aggregate over union all**. All pushdown operations can be enabled based on your requirements.

Table 1-20 Enhanced query of cross-source query

Module	Before Enhancement	After Enhancement
aggregate	The pushdown of **aggregate** is not supported.	• Aggregation functions including **sum**, **avg**, **max**, **min**, and **count** are supported. Example: select count(*) from table • Internal expressions of aggregation functions are supported. Example: select sum(a+b) from table • Calculation of aggregation functions is supported. Example: select avg(a) + max(b) from table • Pushdown of **having** is supported. Example: select sum(a) from table where a>0 group by b having sum(a)>10 • Pushdown of some functions is supported. Pushdown of lines in mathematics, time, and string functions, such as **abs()**, **month()**, and **length()** are supported. In addition to the preceding built-in functions, you can run the **SET** command to add functions supported by data sources. Example: select sum(abs(a)) from table • Pushdown of **limit** and **order by** after **aggregate** is supported. However, the pushdown is not supported in Oracle, because Oracle does not support **limit**. Example: select sum(a) from table where a>0 group by b order by sum(a) limit 5
projection	Only pushdown of simple **projection** is supported. Example: select a, b from table	• Complex expressions can be pushed down. Example: select (a+b)*c from table • Some functions can be pushed down. For details, see the description below the table. Example: select length(a)+abs(b) from table • Pushdown of **limit** and **order by** after **projection** is supported. Example: select a, b+c from table order by a limit 3

Module	Before Enhancement	After Enhancement
predicate	Only simple filtering with the column name on the left of the operator and values on the right is supported. Example: select * from table where a>0 or b in ("aaa", "bbb")	• Complex expression pushdown is supported. Example: select * from table where a +b>c*d or a/c in (1, 2, 3) • Some functions can be pushed down. For details, see the description below the table. Example: select * from table where length(a)>5
aggregate over inner join	Related data from the two tables must be loaded to Spark. The join operation must be performed before the **aggregate** operation.	The following functions are supported: • Aggregation functions including **sum**, **avg**, **max**, **min**, and **count** are supported. • All **aggregate** operations can be performed in a same table. The **group by** operations can be performed on one or two tables and only inner join is supported. The following scenarios are not supported: • **aggregate** cannot be pushed down from both the left- and right-join tables. • **aggregate** contains operations, for example, sum(a+b). • **aggregate** operations, for example, sum(a)+min(b).
aggregate over union all	Related data from the two tables must be loaded to Spark. **union** must be performed before **aggregate**.	Supported scenarios: Aggregation functions including **sum**, **avg**, **max**, **min**, and **count** are supported. Unsupported scenarios: • **aggregate** contains operations, for example, sum(a+b). • **aggregate** operations, for example, sum(a)+min(b).

Precautions

- If external data source is Hive, query operation cannot be performed on foreign tables created by Spark.
- Only MySQL and MPPDB data sources are supported.

1.4.24 Spark2x

1.4.24.1 Basic Principles of Spark2x

NOTE

The Spark2x component applies to MRS 3.x and later versions.

Description

Spark is a memory-based distributed computing framework. In iterative computation scenarios, the computing capability of Spark is 10 to 100 times higher than MapReduce, because data is stored in memory when being processed. Spark can use HDFS as the underlying storage system, enabling users to quickly switch to Spark from MapReduce. Spark provides one-stop data analysis capabilities, such as the streaming processing in small batches, offline batch processing, SQL query, and data mining. Users can seamlessly use these functions in a same application. For details about the new open-source features of Spark2x, see **Spark2x Open Source New Features**.

Features of Spark are as follows:

- Improves the data processing capability through distributed memory computing and directed acyclic graph (DAG) execution engine. The delivered performance is 10 to 100 times higher than that of MapReduce.
- Supports multiple development languages (Scala/Java/Python) and dozens of highly abstract operators to facilitate the construction of distributed data processing applications.
- Builds data processing stacks using **SQL**, **Streaming**, MLlib, and GraphX to provide one-stop data processing capabilities.
- Fits into the Hadoop ecosystem, allowing Spark applications to run on Standalone, Mesos, or Yarn, enabling access of multiple data sources such as HDFS, HBase, and Hive, and supporting smooth migration of the MapReduce application to Spark.

Architecture

Figure 1-104 describes the Spark architecture and **Table 1-21** lists the Spark modules.

Figure 1-104 Spark architecture

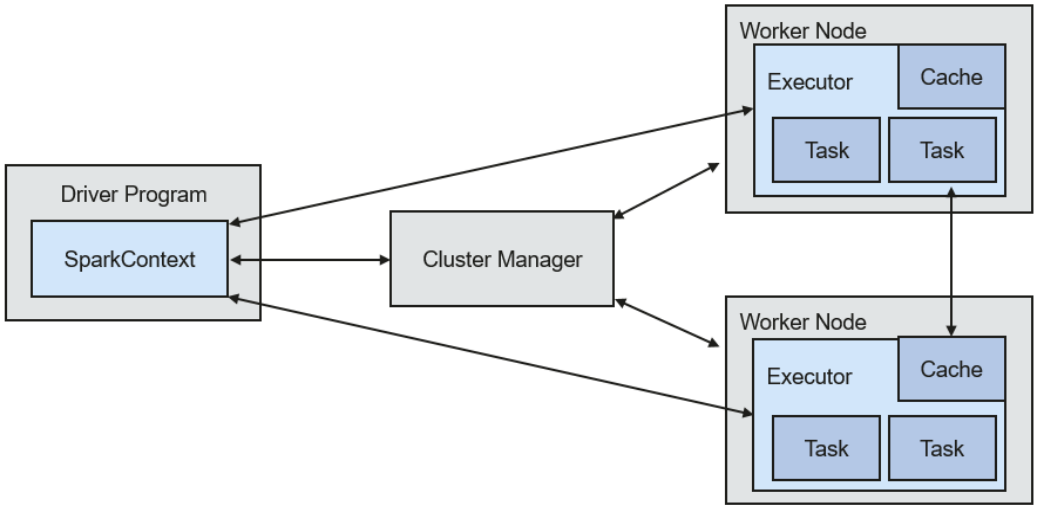


Table 1-21 Basic concepts

Module	Description
Cluster Manager	Cluster manager manages resources in the cluster. Spark supports multiple cluster managers, including Mesos, Yarn, and the Standalone cluster manager that is delivered with Spark. By default, Spark clusters adopt the Yarn cluster manager.
Application	Spark application. It consists of one Driver Program and multiple executors.
Deploy Mode	Deployment in cluster or client mode. In cluster mode, the driver runs on a node inside the cluster. In client mode, the driver runs on the client (outside the cluster).
Driver Program	The main process of the Spark application. It runs the **main()** function of an application and creates SparkContext. It is used for parsing applications, generating stages, and scheduling tasks to executors. Usually, SparkContext represents Driver Program.
Executor	A process started on a Work Node. It is used to execute tasks, and manage and process the data used in applications. A Spark application usually contains multiple executors. Each executor receives commands from the driver and executes one or multiple tasks.
Worker Node	A node that starts and manages executors and resources in a cluster.
Job	A job consists of multiple concurrent tasks. One action operator (for example, a collect operator) maps to one job.
Stage	Each job consists of multiple stages. Each stage is a task set, which is separated by Directed Acyclic Graph (DAG).

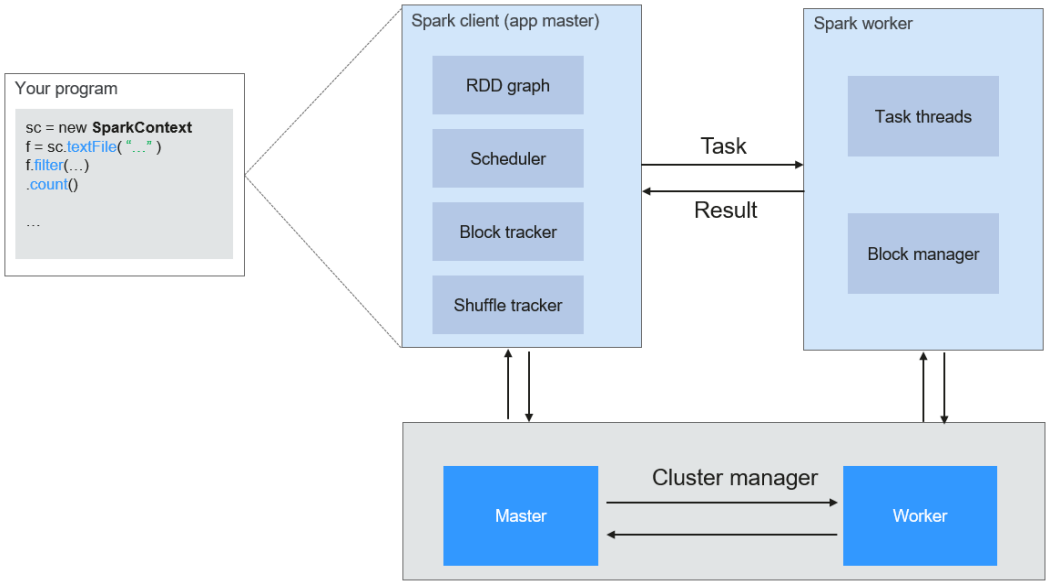
Module	Description
Task	A task carries the computation unit of the service logics. It is the minimum working unit that can be executed on the Spark platform. An application can be divided into multiple tasks based on the execution plan and computation amount.

Spark Principle

Figure 1-105 describes the application running architecture of Spark.

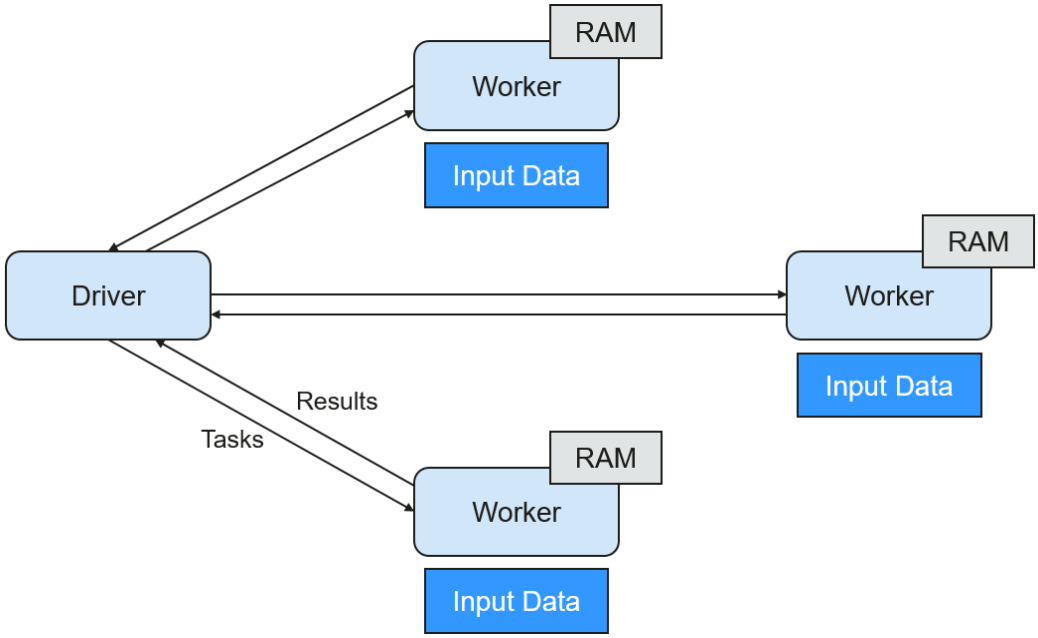
1. An application is running in the cluster as a collection of processes. Driver coordinates the running of the application.
2. To run an application, Driver connects to the cluster manager (such as Standalone, Mesos, and Yarn) to apply for the executor resources, and start ExecutorBackend. The cluster manager schedules resources between different applications. Driver schedules DAGs, divides stages, and generates tasks for the application at the same time.
3. Then, Spark sends the codes of the application (the codes transferred to **SparkContext**, which is defined by JAR or Python) to an executor.
4. After all tasks are finished, the running of the user application is stopped.

Figure 1-105 Spark application running architecture



Spark uses Master and Worker modes, as shown in **Figure 1-106**. A user submits an application on the Spark client, and then the scheduler divides a job into multiple tasks and sends the tasks to each Worker for execution. Each Worker reports the computation results to Driver (Master), and then the Driver aggregates and returns the results to the client.

Figure 1-106 Spark Master-Worker mode



Note the following about the architecture:

- Applications are isolated from each other.

  Each application has an independent executor process, and each executor starts multiple threads to execute tasks in parallel. Each driver schedules its own tasks, and different application tasks run on different JVMs, that is, different executors.
- Different Spark applications do not share data, unless data is stored in the external storage system such as HDFS.
- You are advised to deploy the Driver program in a location that is close to the Worker node because the Driver program schedules tasks in the cluster. For example, deploy the Driver program on the network where the Worker node is located.

Spark on YARN can be deployed in two modes:

- In Yarn-cluster mode, the Spark driver runs inside an ApplicationMaster process which is managed by Yarn in the cluster. After the ApplicationMaster is started, the client can exit without interrupting service running.
- In Yarn-client mode, Driver runs in the client process, and the ApplicationMaster process is used only to apply for requesting resources from Yarn.

Spark Streaming Principle

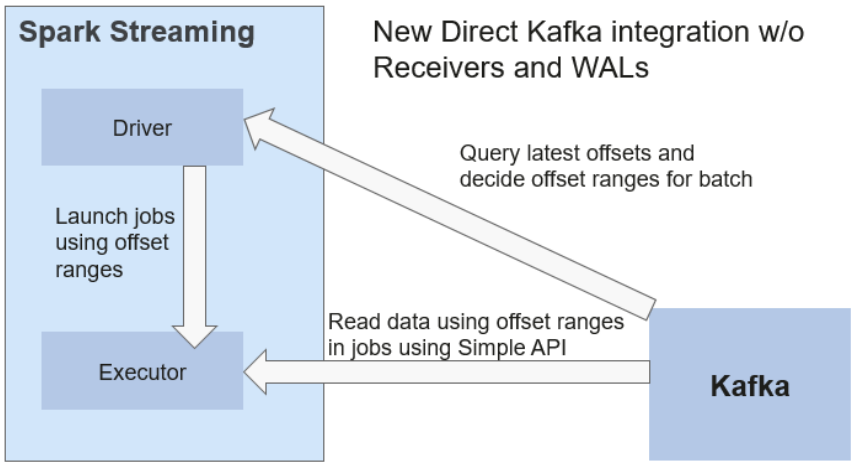
Spark Streaming is a real-time computing framework built on the Spark, which expands the capability for processing massive streaming data. Spark supports two data processing approaches: Direct Streaming and Receiver.

Direct Streaming computing process

In Direct Streaming approach, Direct API is used to process data. Take Kafka Direct API as an example. Direct API provides offset location that each batch range will

read from, which is much simpler than starting a receiver to continuously receive data from Kafka and written data to write-ahead logs (WALs). Then, each batch job is running and the corresponding offset data is ready in Kafka. These offset information can be securely stored in the checkpoint file and read by applications that failed to start.

Figure 1-107 Data transmission through Direct Kafka API



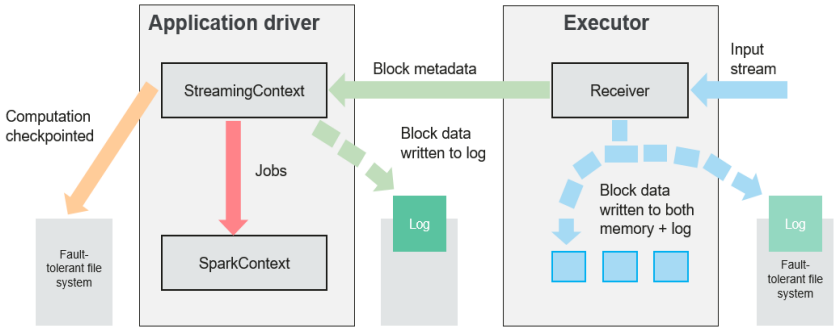
After the failure, Spark Streaming can read data from Kafka again and process the data segment. The processing result is the same no matter Spark Streaming fails or not, because the semantic is processed only once.

Direct API does not need to use the WAL and Receivers, and ensures that each Kafka record is received only once, which is more efficient. In this way, the Spark Streaming and Kafka can be well integrated, making streaming channels be featured with high fault-tolerance, high efficiency, and ease-of-use. Therefore, you are advised to use Direct Streaming to process data.

Receiver computing process

When a Spark Streaming application starts (that is, when the driver starts), the related StreamingContext (the basis of all streaming functions) uses SparkContext to start the receiver to become a long-term running task. These receivers receive and save streaming data to the Spark memory for processing. **Figure 1-108** shows the data transfer lifecycle.

Figure 1-108 Data transfer lifecycle



1. Receive data (blue arrow).

   Receiver divides a data stream into a series of blocks and stores them in the executor memory. In addition, after WAL is enabled, it writes data to the WAL of the fault-tolerant file system.

2. Notify the driver (green arrow).

   The metadata in the received block is sent to StreamingContext in the driver. The metadata includes:

   - Block reference ID used to locate the data position in the Executor memory.
   - Block data offset information in logs (if the WAL function is enabled).

3. Process data (red arrow).

   For each batch of data, StreamingContext uses block information to generate resilient distributed datasets (RDDs) and jobs. StreamingContext executes jobs by running tasks to process blocks in the executor memory.

4. Periodically set checkpoints (orange arrows).
5. For fault tolerance, StreamingContext periodically sets checkpoints and saves them to external file systems.

Fault Tolerance

Spark and its RDD allow seamless processing of failures of any Worker node in the cluster. Spark Streaming is built on top of Spark. Therefore, the Worker node of Spark Streaming also has the same fault tolerance capability. However, Spark Streaming needs to run properly in case of long-time running. Therefore, Spark must be able to recover from faults through the driver process (main process that coordinates all Workers). This poses challenges to the Spark driver fault-tolerance because the Spark driver may be any user application implemented in any computation mode. However, Spark Streaming has internal computation architecture. That is, it periodically executes the same Spark computation in each batch data. Such architecture allows it to periodically store checkpoints to reliable storage space and recover them upon the restart of Driver.

For source data such as files, the Driver recovery mechanism can ensure zero data loss because all data is stored in a fault-tolerant file system such as HDFS. However, for other data sources such as Kafka and Flume, some received data is cached only in memory and may be lost before being processed. This is caused by the distribution operation mode of Spark applications. When the driver process fails, all executors running in the Cluster Manager, together with all data in the memory, are terminated. To avoid such data loss, the WAL function is added to Spark Streaming.

WAL is often used in databases and file systems to ensure persistence of any data operation. That is, first record an operation to a persistent log and perform this operation on data. If the operation fails, the system is recovered by reading the log and re-applying the preset operation. The following describes how to use WAL to ensure persistence of received data:

Receiver is used to receive data from data sources such as Kafka. As a long-time running task in Executor, Receiver receives data, and also confirms received data if supported by data sources. Received data is stored in the Executor memory, and Driver delivers a task to Executor for processing.

After WAL is enabled, all received data is stored to log files in the fault-tolerant file system. Therefore, the received data does not lose even if Spark Streaming

fails. Besides, receiver checks correctness of received data only after the data is pre-written into logs. Data that is cached but not stored can be sent again by data sources after the driver restarts. These two mechanisms ensure zero data loss. That is, all data is recovered from logs or re-sent by data sources.

To enable the WAL function, perform the following operations:

- Set **streamingContext.checkpoint** (path-to-directory) to configure the checkpoint directory, which is an HDFS file path used to store streaming checkpoints and WALs.
- Set **spark.streaming.receiver.writeAheadLog.enable** of SparkConf to **true** (the default value is **false**).

After WAL is enabled, all receivers have the advantage of recovering from reliable received data. You are advised to disable the multi-replica mechanism because the fault-tolerant file system of WAL may also replicate the data.

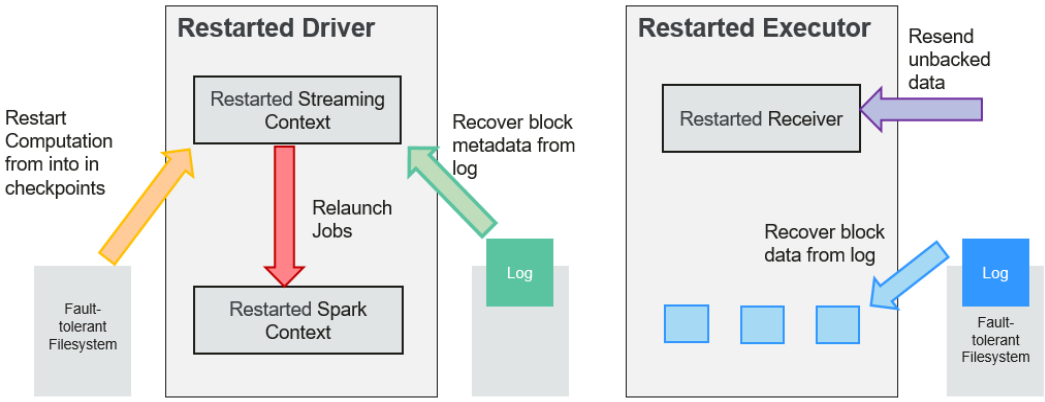
NOTE

The data receiving throughput is lowered after WAL is enabled. All data is written into the fault-tolerant file system. As a result, the write throughput of the file system and the network bandwidth for data replication may become the potential bottleneck. To solve this problem, you are advised to create more receivers to increase the degree of data receiving parallelism or use better hardware to improve the throughput of the fault-tolerant file system.

Recovery Process

When a failed driver is restarted, restart it as follows:

Figure 1-109 Computing recovery process



1. Recover computing. (Orange arrow)

   Use checkpoint information to restart Driver, reconstruct SparkContext and restart Receiver.

2. Recover metadata block. (Green arrow)

   This operation ensures that all necessary metadata blocks are recovered to continue the subsequent computing recovery.

3. Relaunch unfinished jobs. (Red arrow)

   Recovered metadata is used to generate RDDs and corresponding jobs for interrupted batch processing due to failures.

4. Read block data saved in logs. (Blue arrow)

   Block data is directly read from WALs during execution of the preceding jobs, and therefore all essential data reliably stored in logs is recovered.

5. Resend unconfirmed data. (Purple arrow)

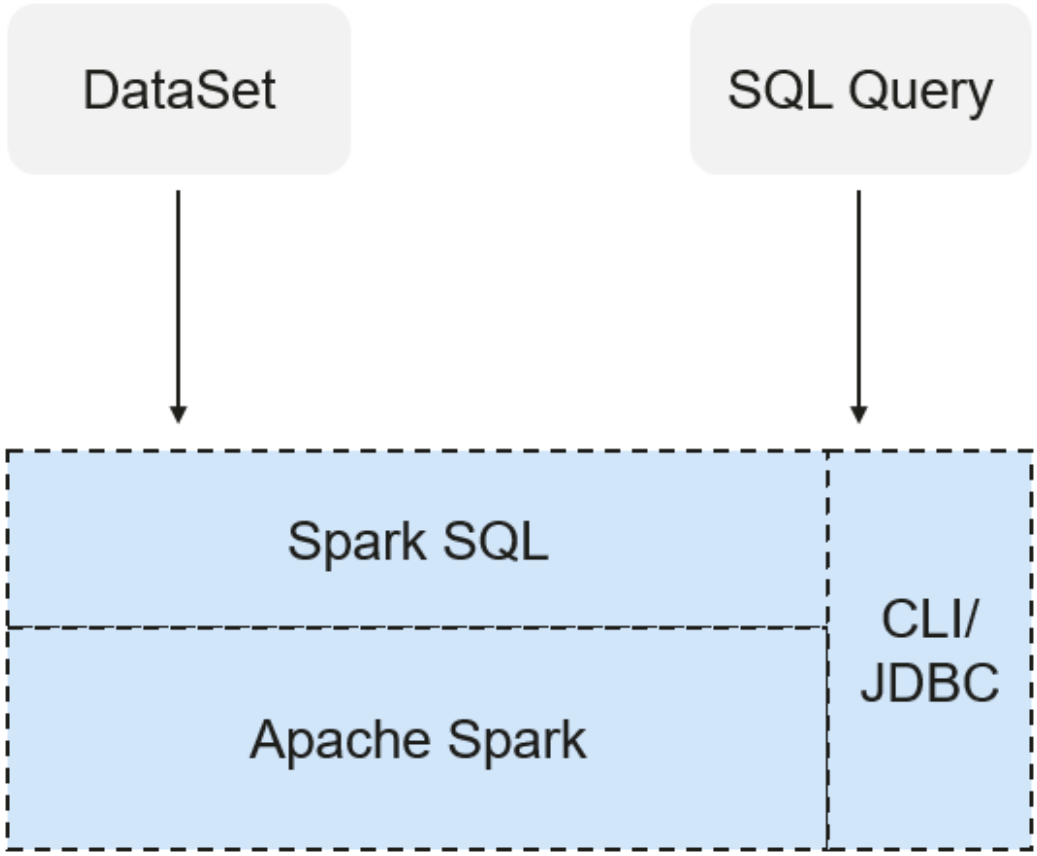
   Data that is cached but not stored to logs upon failures is re-sent by data sources, because the receiver does not confirm the data.

Therefore, by using WALs and reliable Receiver, Spark Streaming can avoid input data loss caused by Driver failures.

SparkSQL and DataSet Principle

SparkSQL

Figure 1-110 SparkSQL and DataSet



Spark SQL is a module for processing structured data. In Spark application, SQL statements or DataSet APIs can be seamlessly used for querying structured data.

Spark SQL and DataSet also provide a universal method for accessing multiple data sources such as Hive, CSV, Parquet, ORC, JSON, and JDBC. These data sources also allow data interaction. Spark SQL reuses the Hive frontend processing logic and metadata processing module. With the Spark SQL, you can directly query existing Hive data.

In addition, Spark SQL also provides API, CLI, and JDBC APIs, allowing diverse accesses to the client.

Spark SQL Native DDL/DML

In Spark 1.5, lots of Data Definition Language (DDL)/Data Manipulation Language (DML) commands are pushed down to and run on the Hive, causing coupling with the Hive and inflexibility such as unexpected error reports and results.

Spark2x realizes command localization and replaces the Hive with Spark SQL Native DDL/DML to run DDL/DML commands. Additionally, the decoupling from the Hive is realized and commands can be customized.

DataSet

A DataSet is a strongly typed collection of domain-specific objects that can be transformed in parallel using functional or relational operations. Each Dataset also has an untyped view called a DataFrame, which is a Dataset of Row.

The DataFrame is a structured and distributed dataset consisting of multiple columns. The DataFrame is equal to a table in the relationship database or the DataFrame in the R/Python. The DataFrame is the most basic concept in the Spark SQL, which can be created by using multiple methods, such as the structured dataset, Hive table, external database or RDD.

Operations available on DataSets are divided into transformations and actions.

- A transformation operation can generate a new DataSet,
  for example, **map**, **filter**, **select**, and **aggregate (groupBy)**.
- An action operation can trigger computation and return results,
  for example, **count**, **show**, or write data to the file system.

You can use either of the following methods to create a DataSet:

- The most common way is by pointing Spark to some files on storage systems, using the **read** function available on a SparkSession.
  ```
  val people = spark.read.parquet("...").as[Person]  // Scala
  DataSet<Person> people = spark.read().parquet("...").as(Encoders.bean(Person.class));//Java
  ```
- You can also create a DataSet using the transformation operation available on an existing one. For example, apply the map operation on an existing DataSet to create a DataSet:
  ```
  val names = people.map(_.name) // In Scala: names is Dataset.
  Dataset<String> names = people.map((Person p) -> p.name, Encoders.STRING)); // Java
  ```

CLI and JDBCServer

In addition to programming APIs, Spark SQL also provides the CLI/JDBC APIs.

- Both **spark-shell** and **spark-sql** scripts can provide the CLI for debugging.
- JDBCServer provides JDBC APIs. External systems can directly send JDBC requests to calculate and parse structured data.

SparkSession Principle

SparkSession is a unified API in Spark2x and can be regarded as a unified entry for reading data. SparkSession provides a single entry point to perform many operations that were previously scattered across multiple classes, and also provides accessor methods to these older classes to maximize compatibility.

A SparkSession can be created using a builder pattern. The builder will automatically reuse the existing SparkSession if there is a SparkSession; or create

a SparkSession if it does not exist. During I/O transactions, the configuration item settings in the builder are automatically synchronized to Spark and Hadoop.

```
import org.apache.spark.sql.SparkSession
val sparkSession = SparkSession.builder
  .master("local")
  .appName("my-spark-app")
  .config("spark.some.config.option", "config-value")
  .getOrCreate()
```

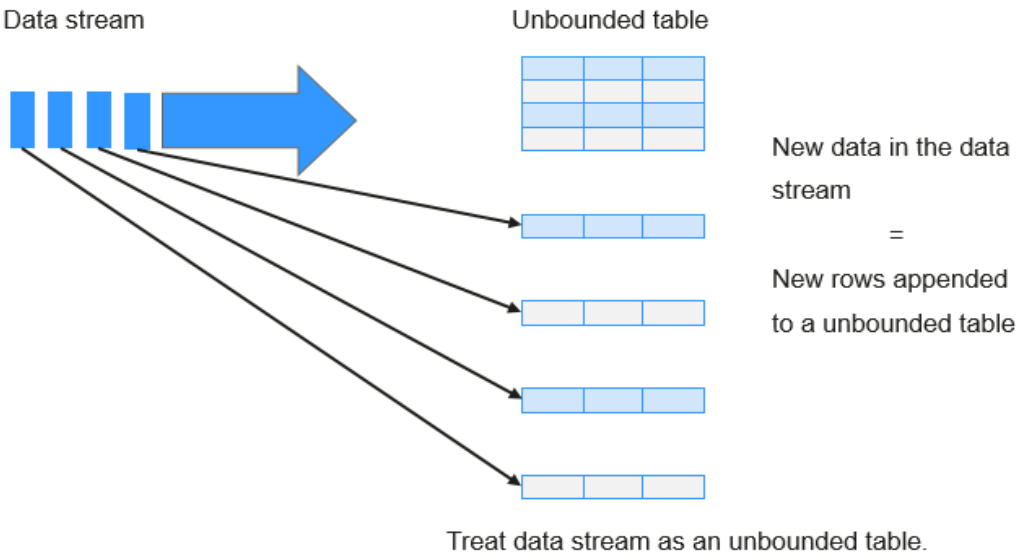
- SparkSession can be used to execute SQL queries on data and return results as DataFrame.
  ```
  sparkSession.sql("select * from person").show
  ```
- SparkSession can be used to set configuration items during running. These configuration items can be replaced with variables in SQL statements.
  ```
  sparkSession.conf.set("spark.some.config", "abcd")
  sparkSession.conf.get("spark.some.config")
  sparkSession.sql("select ${spark.some.config}")
  ```
- SparkSession also includes a "catalog" method that contains methods to work with Metastore (data catalog). After this method is used, a dataset is returned, which can be run using the same Dataset API.
  ```
  val tables = sparkSession.catalog.listTables()
  val columns = sparkSession.catalog.listColumns("myTable")
  ```
- Underlying SparkContext can be accessed by SparkContext API of SparkSession.
  ```
  val sparkContext = sparkSession.sparkContext
  ```

Structured Streaming Principle

Structured Streaming is a stream processing engine built on the Spark SQL engine. You can use the Dataset/DataFrame API in Scala, Java, Python, or R to express streaming aggregations, event-time windows, and stream-stream joins. If streaming data is incrementally and continuously produced, Spark SQL will continue to process the data and synchronize the result to the result set. In addition, the system ensures end-to-end exactly-once fault-tolerance guarantees through checkpoints and WALs.

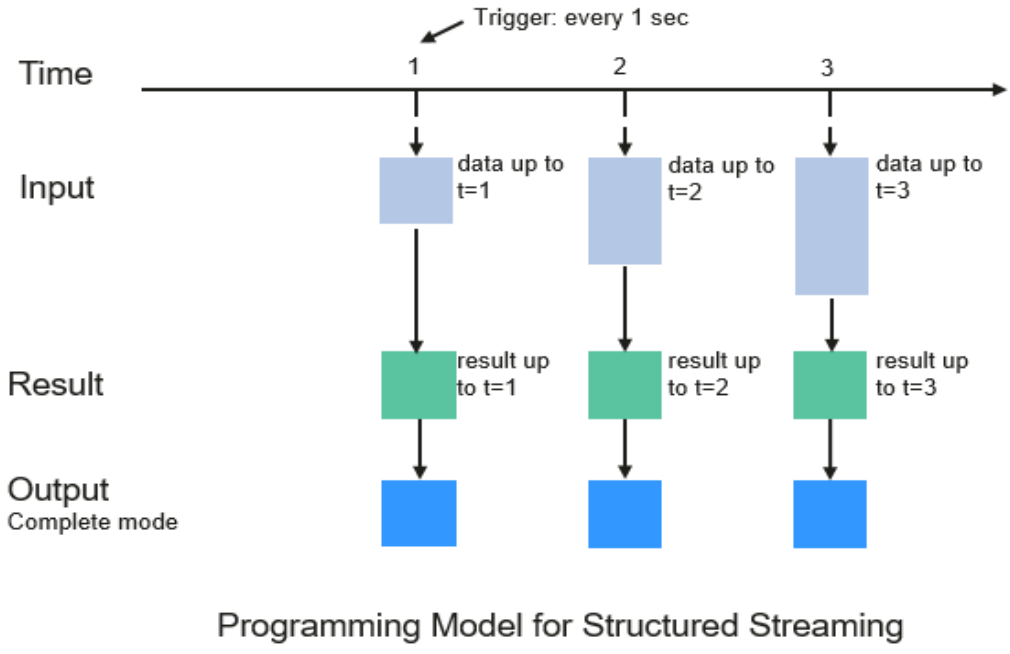
The core of Structured Streaming is to take streaming data as an incremental database table. Similar to the data block processing model, the streaming data processing model applies query operations on a static database table to streaming computing, and Spark uses standard SQL statements for query, to obtain data from the incremental and unbounded table.

Figure 1-111 Unbounded table of Structured Streaming



Each query operation will generate a result table. At each trigger interval, updated data will be synchronized to the result table. Whenever the result table is updated, the updated result will be written into an external storage system.

Figure 1-112 Structured Streaming data processing model



Storage modes of Structured Streaming at the output phase are as follows:

- Complete Mode: The updated result sets are written into the external storage system. The write operation is performed by a connector of the external storage system.

- Append Mode: If an interval is triggered, only added data in the result table will be written into an external system. This is applicable only on the queries where existing rows in the result table are not expected to change.
- Update Mode: If an interval is triggered, only updated data in the result table will be written into an external system, which is the difference between the Complete Mode and Update Mode.

Concepts

- **RDD**

  Resilient Distributed Dataset (RDD) is a core concept of Spark. It indicates a read-only and partitioned distributed dataset. Partial or all data of this dataset can be cached in the memory and reused between computations.

  RDD Creation

  - An RDD can be created from the input of HDFS or other storage systems that are compatible with Hadoop.
  - A new RDD can be converted from a parent RDD.
  - An RDD can be converted from a collection of datasets through encoding.

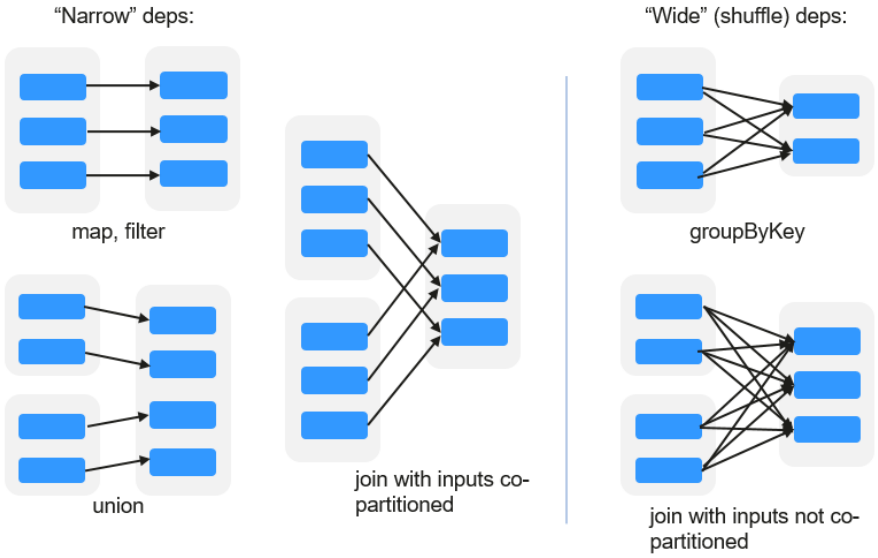
  RDD Storage

  - You can select different storage levels to store an RDD for reuse. (There are 11 storage levels to store an RDD.)
  - By default, the RDD is stored in the memory. When the memory is insufficient, the RDD overflows to the disk.

- **RDD Dependency**

  The RDD dependency includes the narrow dependency and wide dependency.

  Figure 1-113 RDD dependency

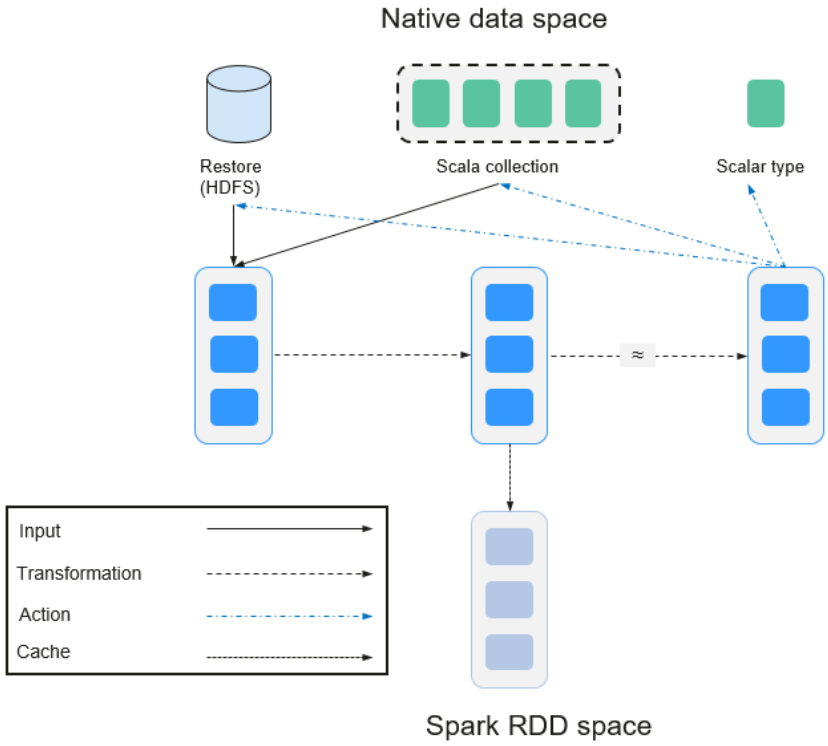
  - **Narrow dependency**: Each partition of the parent RDD is used by at most one partition of the child RDD.
  - **Wide dependency**: Partitions of the child RDD depend on all partitions of the parent RDD.

The narrow dependency facilitates the optimization. Logically, each RDD operator is a fork/join (the join is not the join operator mentioned above but the barrier used to synchronize multiple concurrent tasks); fork the RDD to each partition, and then perform the computation. After the computation, join the results, and then perform the fork/join operation on the next RDD operator. It is uneconomical to directly translate the RDD into physical implementation. The first is that every RDD (even intermediate result) needs to be physicalized into memory or storage, which is time-consuming and occupies much space. The second is that as a global barrier, the join operation is very expensive and the entire join process will be slowed down by the slowest node. If the partitions of the child RDD narrowly depend on that of the parent RDD, the two fork/join processes can be combined to implement classic fusion optimization. If the relationship in the continuous operator sequence is narrow dependency, multiple fork/join processes can be combined to reduce a large number of global barriers and eliminate the physicalization of many RDD intermediate results, which greatly improves the performance. This is called pipeline optimization in Spark.

- **Transformation and Action (RDD Operations)**

  Operations on RDD include transformation (the return value is an RDD) and action (the return value is not an RDD). **Figure 1-114** shows the RDD operation process. The transformation is lazy, which indicates that the transformation from one RDD to another RDD is not immediately executed. Spark only records the transformation but does not execute it immediately. The real computation is started only when the action is started. The action returns results or writes the RDD data into the storage system. The action is the driving force for Spark to start the computation.

Figure 1-114 RDD operation



The data and operation model of RDD are quite different from those of Scala.

```
val file = sc.textFile("hdfs://...")
val errors = file.filter(_.contains("ERROR"))
errors.cache()
errors.count()
```

a. The textFile operator reads log files from the HDFS and returns files (as an RDD).

b. The filter operator filters rows with **ERROR** and assigns them to errors (a new RDD). The filter operator is a transformation.

c. The cache operator caches errors for future use.

d. The count operator returns the number of rows of errors. The count operator is an action.

Transformation includes the following types:

– The RDD elements are regarded as simple elements.

  The input and output has the one-to-one relationship, and the partition structure of the result RDD remains unchanged, for example, map.

  The input and output has the one-to-many relationship, and the partition structure of the result RDD remains unchanged, for example, flatMap (one element becomes a sequence containing multiple elements after map and then flattens to multiple elements).

  The input and output has the one-to-one relationship, but the partition structure of the result RDD changes, for example, union (two RDDs integrates to one RDD, and the number of partitions becomes the sum of the number of partitions of two RDDs) and coalesce (partitions are reduced).

  Operators of some elements are selected from the input, such as filter, distinct (duplicate elements are deleted), subtract (elements only exist in this RDD are retained), and sample (samples are taken).

– The RDD elements are regarded as key-value pairs.

  Perform the one-to-one calculation on the single RDD, such as mapValues (the partition mode of the source RDD is retained, which is different from map).

  Sort the single RDD, such as sort and partitionBy (partitioning with consistency, which is important to the local optimization).

  Restructure and reduce the single RDD based on key, such as groupByKey and reduceByKey.

  Join and restructure two RDDs based on the key, such as join and cogroup.

  > NOTE
  >
  > The later three operations involving sorting are called shuffle operations.

Action includes the following types:

– Generate scalar configuration items, such as **count** (the number of elements in the returned RDD), **reduce**, **fold/aggregate** (the number of scalar configuration items that are returned), and **take** (the number of elements before the return).

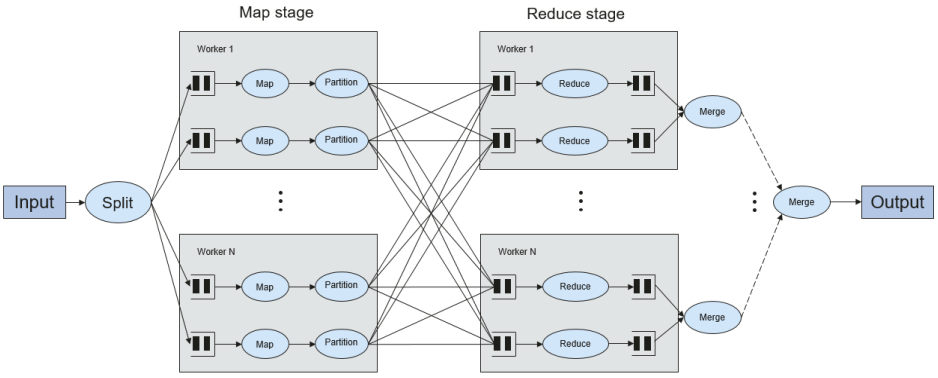
  - Generate the Scala collection, such as **collect** (import all elements in the RDD to the Scala collection) and **lookup** (look up all values corresponds to the key).
  - Write data to the storage, such as **saveAsTextFile** (which corresponds to the preceding **textFile**).
  - Check points, such as the **checkpoint** operator. When Lineage is quite long (which occurs frequently in graphics computation), it takes a long period of time to execute the whole sequence again when a fault occurs. In this case, checkpoint is used as the check point to write the current data to stable storage.

- **Shuffle**

  Shuffle is a specific phase in the MapReduce framework, which is located between the Map phase and the Reduce phase. If the output results of Map are to be used by Reduce, the output results must be hashed based on a key and distributed to each Reducer. This process is called Shuffle. Shuffle involves the read and write of the disk and the transmission of the network, so that the performance of Shuffle directly affects the operation efficiency of the entire program.

  The figure below shows the entire process of the MapReduce algorithm.

  Figure 1-115 Algorithm process



  Shuffle is a bridge to connect data. The following describes the implementation of shuffle in Spark.

  Shuffle divides a job of Spark into multiple stages. The former stages contain one or more ShuffleMapTasks, and the last stage contains one or more ResultTasks.

- **Spark Application Structure**

  The Spark application structure includes the initialized SparkContext and the main program.

  - Initialized SparkContext: constructs the operating environment of the Spark Application.

    Constructs the SparkContext object. The following is an example:

    ```
    new SparkContext(master, appName, [SparkHome], [jars])
    ```

    Parameter description:

    master: indicates the link string. The link modes include local, Yarn-cluster, and Yarn-client.

appName: indicates the application name.

SparkHome: indicates the directory where Spark is installed in the cluster.

jars: indicates the code and dependency package of an application.

– Main program: processes data.

For details about how to submit an application, visit **https://archive.apache.org/dist/spark/docs/3.1.1/submitting-applications.html**.

- **Spark Shell Commands**

  The basic Spark shell commands support the submission of Spark applications. The Spark shell commands are as follows:

```
./bin/spark-submit \
  --class <main-class> \
  --master <master-url> \
  ... # other options
  <application-jar> \
  [application-arguments]
```

  Parameter description:

  --class: indicates the name of the class of a Spark application.

  --master: indicates the master to which the Spark application links, such as Yarn-client and Yarn-cluster.

  application-jar: indicates the path of the JAR file of the Spark application.

  application-arguments: indicates the parameter required to submit the Spark application. This parameter can be left blank.

- **Spark JobHistory Server**

  The Spark web UI is used to monitor the details in each phase of the Spark framework of a running or historical Spark job and provide the log display, which helps users to develop, configure, and optimize the job in more fine-grained units.

1.4.24.2 Spark2x HA Solution

1.4.24.2.1 Spark2x Multi-active Instance

Background

Based on existing JDBCServers in the community, multi-active-instance HA is used to achieve the high availability. In this mode, multiple JDBCServers coexist in the cluster and the client can randomly connect any JDBCServer to perform service operations. When one or multiple JDBCServers stop working, a client can connect to another normal JDBCServer.

Compared with active/standby HA, multi-active instance HA eliminates the following restrictions:

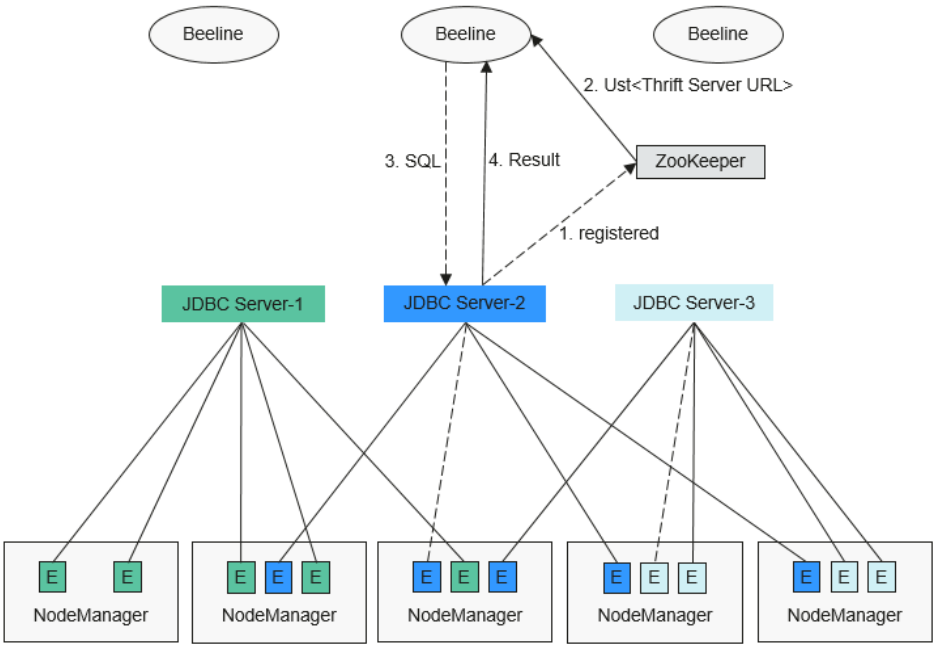
- In active/standby HA, when the active/standby switchover occurs, the unavailable period cannot be controlled by JDBCServer, but determined by Yarn service resources.
- In Spark, the Thrift JDBC similar to HiveServer2 provides services and users access services through Beeline and JDBC API. Therefore, the processing capability of the JDBCServer cluster depends on the single-point capability of the primary server, and the scalability is insufficient.

Multi-active instance HA not only prevents service interruption caused by switchover, but also enables cluster scale-out to secure high concurrency.

Implementation

The following figure shows the basic principle of multi-active instance HA of Spark JDBCServer.

Figure 1-116 Spark JDBCServer HA



1. After JDBCServer is started, it registers with ZooKeeper by writing node information in a specified directory. Node information includes the JDBCServer instance IP, port number, version, and serial number (information of different nodes is separated by commas).

   An example is provided as follows:

   ```
   [serverUri=192.168.169.84:22550
   ;version=8.1.2;sequence=0000001244,serverUri=192.168.195.232:22550 ;version=8.1.2;sequence=00000
   01242,serverUri=192.168.81.37:22550 ;version=8.1.2;sequence=0000001243]
   ```

2. To connect to JDBCServer, the client must specify the namespace, which is the directory of JDBCServer instances in ZooKeeper. During the connection, a JDBCServer instance is randomly selected from the specified namespace. For details about URL, see **URL Connection**.
3. After the connection succeeds, the client sends SQL statements to JDBCServer.
4. JDBCServer executes received SQL statements and sends results back to the client.

In multi-active instance HA mode, all JDBCServer instances are independent and equivalent. When one instance is interrupted during upgrade, other JDBCServer instances can accept the connection request from the client.

Following rules must be followed in the multi-active instance HA of Spark JDBCServer:

- If a JDBCServer instance exits abnormally, no other instance will take over the sessions and services running on this abnormal instance.
- When the JDBCServer process is stopped, corresponding nodes are deleted from ZooKeeper.
- The client randomly selects the server, which may result in uneven session allocation, and finally result in imbalance of instance load.
- After the instance enters the maintenance mode (in which no new connection request from the client is accepted), services still running on the instance may fail when the decommissioning times out.

URL Connection

Multi-active instance mode

In multi-active instance mode, the client reads content from the ZooKeeper node and connects to JDBCServer. The connection strings are as follows:

- Security mode:
  - If Kinit authentication is enabled, the JDBCURL is as follows:
    jdbc:hive2://<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.**<*System domain name*>**@**<*System domain name*>**;**

    NOTE

    - **<zkNode_IP>:<zkNode_Port>** indicates the ZooKeeper URL. Use commas (,) to separate multiple URLs,

      For example, **192.168.81.37:2181,192.168.195.232:2181,192.168.169.84:2181**.
    - **sparkthriftserver2x** indicates the directory in ZooKeeper, where a random JDBCServer instance is connected to the client.

    For example, when you use Beeline client for connection in security mode, run the following command:

    sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://**<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.**<*System domain name*>**@**<*System domain name*>**;"**
  - If Keytab authentication is enabled, the JDBCURL is as follows:
    jdbc:hive2://<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.**<*System domain name*>**@**<*System domain name*>**;user.principal=**<*principal_name*>**;user.keytab=**<*path_to_keytab*>

    <*principal_name*> indicates the principal of Kerberos user, for example, **test@**<*System domain name*>. <*path_to_keytab*> indicates the Keytab file path corresponding to <*principal_name*>, for example, **/opt/auth/test/user.keytab**.
- Common mode:
  jdbc:hive2://<*zkNode1_IP*>**:**<*zkNode1_Port*>**,**<*zkNode2_IP*>**:**<*zkNode2_Port*>**,**<*zkNode3_IP*>**:**<*zkNode3_Port*>**/**;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=**sparkthriftserver2x**;

For example, when you use Beeline client for connection in common mode, run the following command:

sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://***<zkNode1_IP>:<zkNode1_Port>,<zkNode2_IP>:<zkNode2_Port>,<zkNode3_IP>:<zkNode3_Port>*/**;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;"**

Non-multi-active instance mode

In non-multi-active instance mode, a client connects to a specified JDBCServer node. Compared with multi-active instance mode, the connection string in non-multi-active instance mode does not contain **serviceDiscoveryMode** and **zooKeeperNamespace** parameters about ZooKeeper.

For example, when you use Beeline client to connect JDBCServer in non-multi-active instance mode, run the following command:

sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://***<server_IP>:<server_Port>*/**;user.principal=spark2x/hadoop.***<System domain name>*@*<System domain name>*;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.***<System domain name>*@*<System domain name>*;"**

NOTE

- **<server_IP>:<server_Port>** indicates the URL of the specified JDBCServer node.
- **CLIENT_HOME** indicates the client path.

Except the connection method, operations of JDBCServer API in multi-active instance mode and non-multi-active instance mode are the same. Spark JDBCServer is another implementation of HiveServer2 in Hive.

1.4.24.2.2 Spark2x Multi-tenant

Background

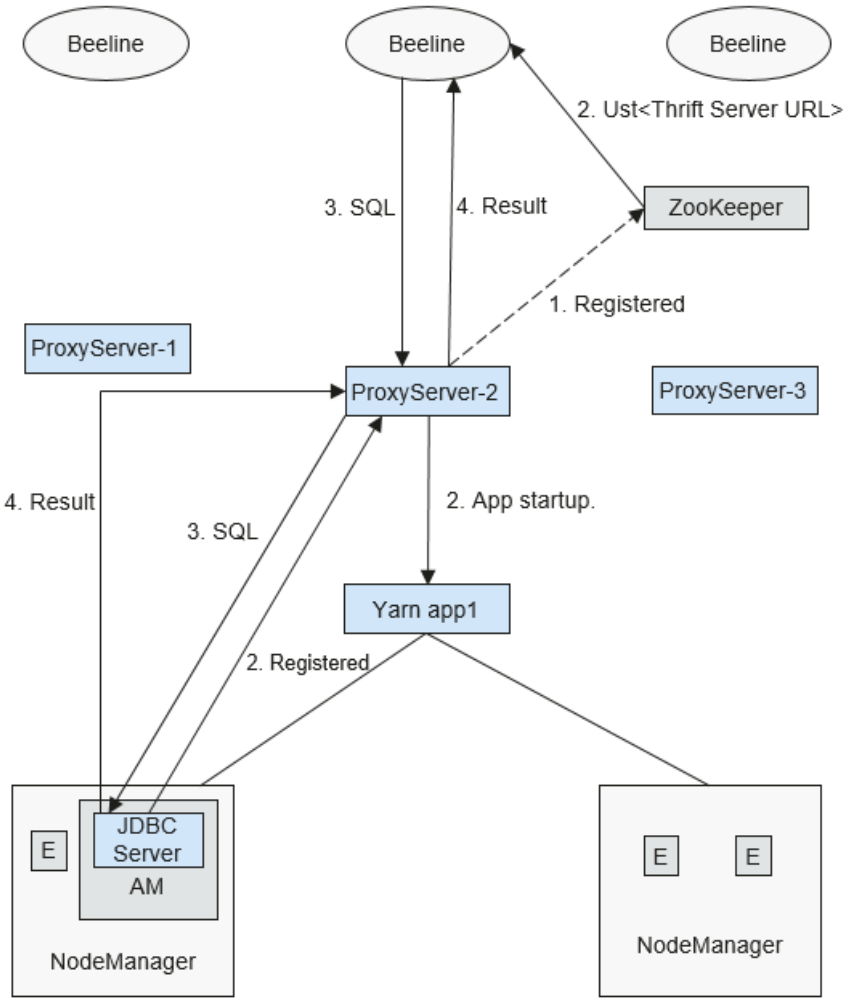
In the JDBCServer multi-active instance mode, JDBCServer implements the Yarn-client mode but only one Yarn resource queue is available. To solve the resource limitation problem, the multi-tenant mode is introduced.

In multi-tenant mode, JDBCServers are bound with tenants. Each tenant corresponds to one or more JDBCServers, and a JDBCServer provides services for only one tenant. Different tenants can be configured with different Yarn queues to implement resource isolation. In addition, JDBCServer can be dynamically started as required to avoid resource waste.

Implementation

Figure 1-117 shows the HA solution of the multi-tenant mode.

Figure 1-117 Multi-tenant mode of Spark JDBCServer



1. When ProxyServer is started, it registers with ZooKeeper by writing node information in a specified directory. Node information includes the instance IP, port number, version, and serial number (information of different nodes is separated by commas).

   NOTE

   In multi-tenant mode, the JDBCServer instance on MRS page indicates ProxyServer, the JDBCServer agent.

   An example is provided as follows:

   ```
   serverUri=192.168.169.84:22550
   ;version=8.1.2;sequence=0000001244,serverUri=192.168.195.232:22550
   ;version=8.1.2;sequence=0000001242,serverUri=192.168.81.37:22550
   ;version=8.1.2;sequence=0000001243,
   ```

2. To connect to ProxyServer, the client must specify a namespace, which is the directory of the ProxyServer instance that you want to access in ZooKeeper. When the client connects to ProxyServer, an instance under Namespace is randomly selected for connection. For details about the URL, see **URL Connection**.
3. After the client successfully connects to ProxyServer, ProxyServer checks whether the JDBCServer of a tenant exists. If yes, Beeline connects the

JDBCServer. If no, a new JDBCServer is started in Yarn-cluster mode. After the startup of JDBCServer, ProxyServer obtains the IP address of the JDBCServer and establishes the connection between Beeline and JDBCServer.

4. The client sends SQL statements to ProxyServer, which then forwards statements to the connected JDBCServer. JDBCServer returns the results to ProxyServer, which then returns the results to the client.

In multi-tenant HA mode, all ProxyServer instances are independent and equivalent. If one instance is interrupted during upgrade, other instances can accept the connection request from the client.

URL Connection

Multi-tenant mode

In multi-tenant mode, the client reads content from the ZooKeeper node and connects to ProxyServer. The connection strings are as follows:

- Security mode:
  - If Kinit authentication is enabled, the client URL is as follows:
    jdbc:hive2://<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.**<*System domain name*>@<*System domain name*>;

    NOTE

    - **<zkNode_IP>:<zkNode_Port>** indicates the ZooKeeper URL. Use commas (,) to separate multiple URLs,

      For example, **192.168.81.37:2181,192.168.195.232:2181,192.168.169.84:2181**.
    - **sparkthriftserver2x** indicates the ZooKeeper directory, where a random JDBCServer instance is connected to the client.

    For example, when you use Beeline client for connection in security mode, run the following command:

    sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://**<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.**<*System domain name*>**@**<*System domain name*>**;"**
  - If Keytab authentication is enabled, the URL is as follows:
    jdbc:hive2://<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>**/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.**<*System domain name*>@<*System domain name*>**;user.principal=**<*principal_name*>**;user.keytab=**<*path_to_keytab*>

    <*principal_name*> indicates the principal of Kerberos user, for example, **test@**<*System domain name*>. <*path_to_keytab*> indicates the Keytab file path corresponding to <*principal_name*>, for example, **/opt/auth/test/user.keytab**.
- Common mode:
  jdbc:hive2://<*zkNode1_IP*>:<*zkNode1_Port*>,<*zkNode2_IP*>:<*zkNode2_Port*>,<*zkNode3_IP*>:<*zkNode3_Port*>|;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=**sparkthriftserver2x**;

For example, when you use Beeline client for connection in common mode, run the following command:

sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://***<zkNode1_IP>:<zkNode1_Port>,<zkNode2_IP>:<zkNode2_Port>,<zkNode3_IP>:<zkNode3_Port>*/**;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;"**

Non-multi-tenant mode

In non-multi-tenant mode, a client connects to a specified JDBCServer node. Compared with multi-active instance mode, the connection string in non-multi-active instance mode does not contain **serviceDiscoveryMode** and **zooKeeperNamespace** parameters about ZooKeeper.

For example, when you use Beeline client to connect JDBCServer in non-multi-tenant instance mode, run the following command:

sh *CLIENT_HOME*/**spark/bin/beeline -u "jdbc:hive2://***<server_IP>:<server_Port>*/**;user.principal=spark/hadoop.***<System domain name>@<System domain name>***;saslQop=auth-conf;auth=KERBEROS;principal=spark/hadoop.***<System domain name>@<System domain name>***;"**

NOTE

- **<server_IP>:<server_Port>** indicates the URL of the specified JDBCServer node.
- **CLIENT_HOME** indicates the client path.

Except the connection method, other operations of JDBCServer API in multi-tenant mode and non-multi-tenant mode are the same. Spark JDBCServer is another implementation of HiveServer2 in Hive.

Specifying a Tenant

Generally, the client submitted by a user connects to the default JDBCServer of the tenant to which the user belongs. If you want to connect the client to the JDBCServer of a specified tenant, add the **--hiveconf mapreduce.job.queuename** parameter.

Command for connecting Beeline is as follows (**aaa** indicates the tenant name):

beeline --hiveconf mapreduce.job.queuename=aaa -u 'jdbc:hive2://192.168.39.30:2181,192.168.40.210:2181,192.168.215.97:2181;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.*<System domain name>@<System domain name>*;'

1.4.24.3 Relationship Between Spark2x and Other Components

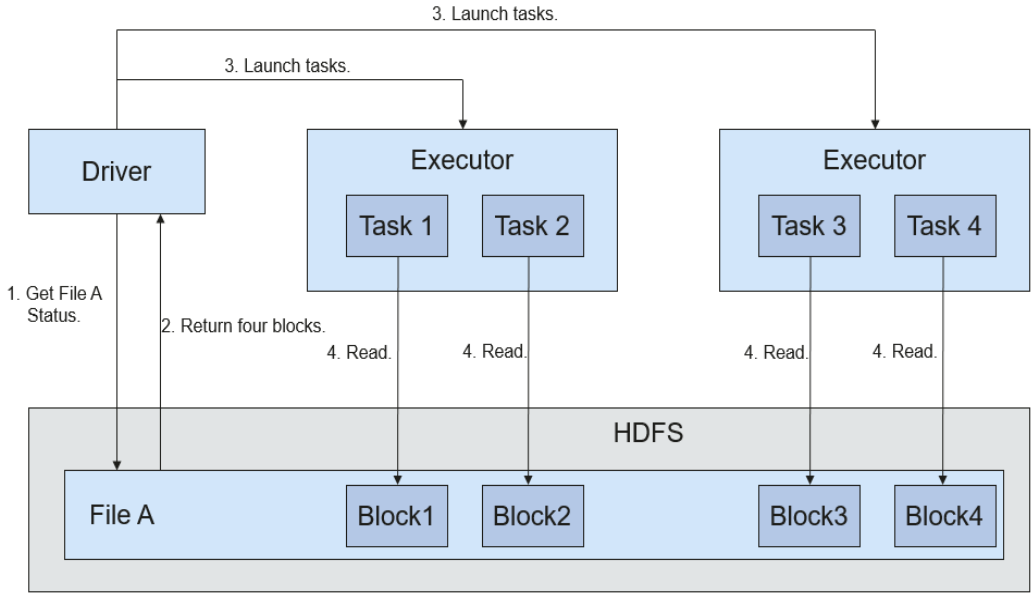
Relationship Between Spark and HDFS

Data computed by Spark comes from multiple data sources, such as local files and HDFS. Most data comes from HDFS which can read data in large scale for parallel computing After being computed, data can be stored in HDFS.

Spark involves Driver and Executor. Driver schedules tasks and Executor runs tasks.

Figure 1-118 describes the file reading process.

Figure 1-118 File reading process

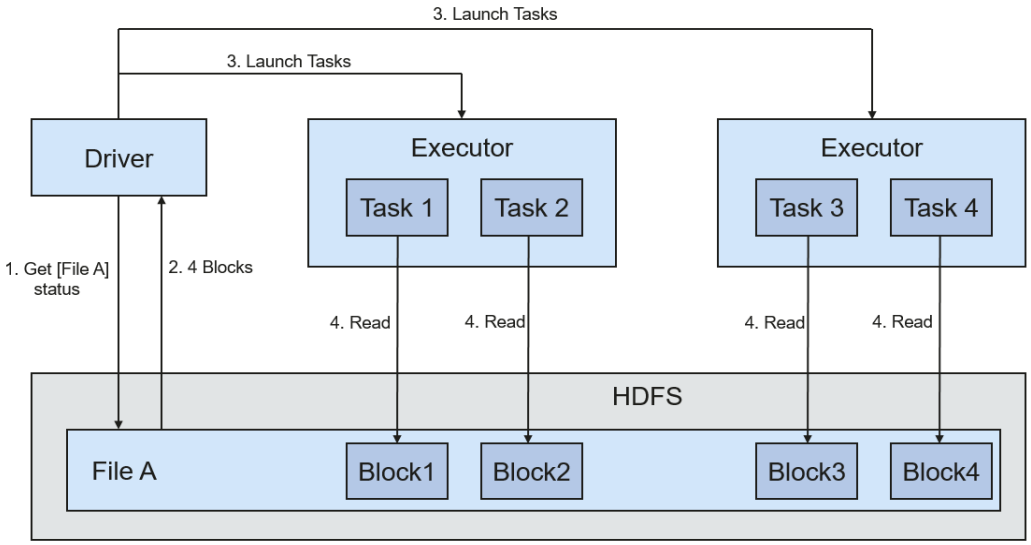


The file reading process is as follows:

1. Driver interconnects with HDFS to obtain the information of File A.
2. The HDFS returns the detailed block information about this file.
3. Driver sets a parallel degree based on the block data amount, and creates multiple tasks to read the blocks of this file.
4. Executor runs the tasks and reads the detailed blocks as part of the Resilient Distributed Dataset (RDD).

Figure 1-119 describes the file writing process.

Figure 1-119 File writing process



The file writing process is as follows:

1. Driver creates a directory where the file is to be written.
2. Based on the RDD distribution status, the number of tasks related to data writing is computed, and these tasks are sent to Executor.
3. Executor runs these tasks, and writes the RDD data to the directory created in 1.

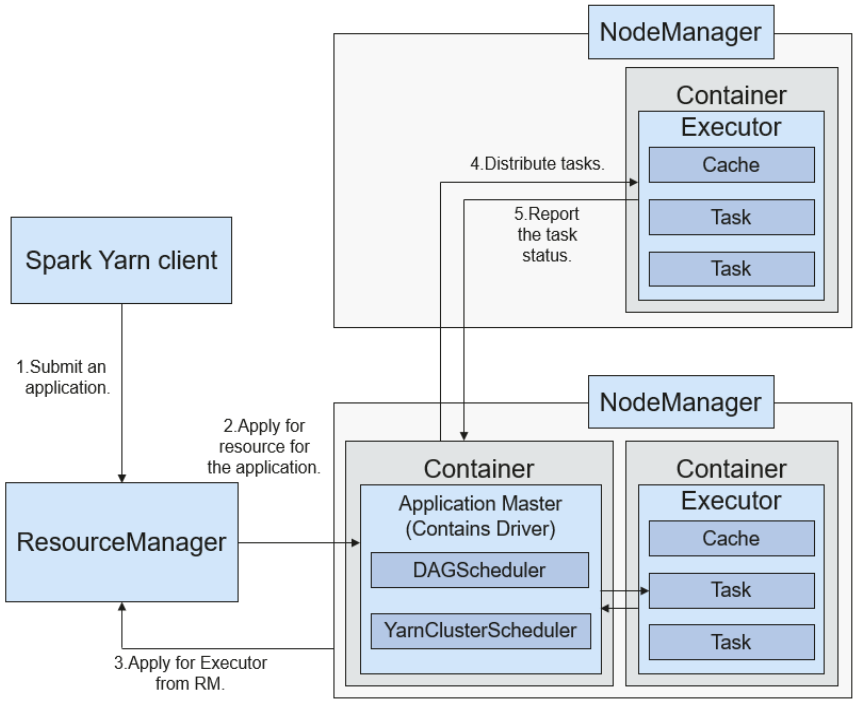
Relationship with Yarn

The Spark computing and scheduling can be implemented using Yarn mode. Spark enjoys the computing resources provided by Yarn clusters and runs tasks in a distributed way. Spark on Yarn has two modes: Yarn-cluster and Yarn-client.

- Yarn-cluster mode

  Figure 1-120 describes the operation framework.

  Figure 1-120 Spark on Yarn-cluster operation framework



  Spark on Yarn-cluster implementation process:

  a. The client generates the application information, and then sends the information to ResourceManager.
  b. ResourceManager allocates the first container (ApplicationMaster) to SparkApplication and starts the driver on the container.
  c. ApplicationMaster applies for resources from ResourceManager to run the container.

     ResourceManager allocates the containers to ApplicationMaster, which communicates with the related NodeManagers and starts the executor in the obtained container. After the executor is started, it registers with drivers and applies for tasks.

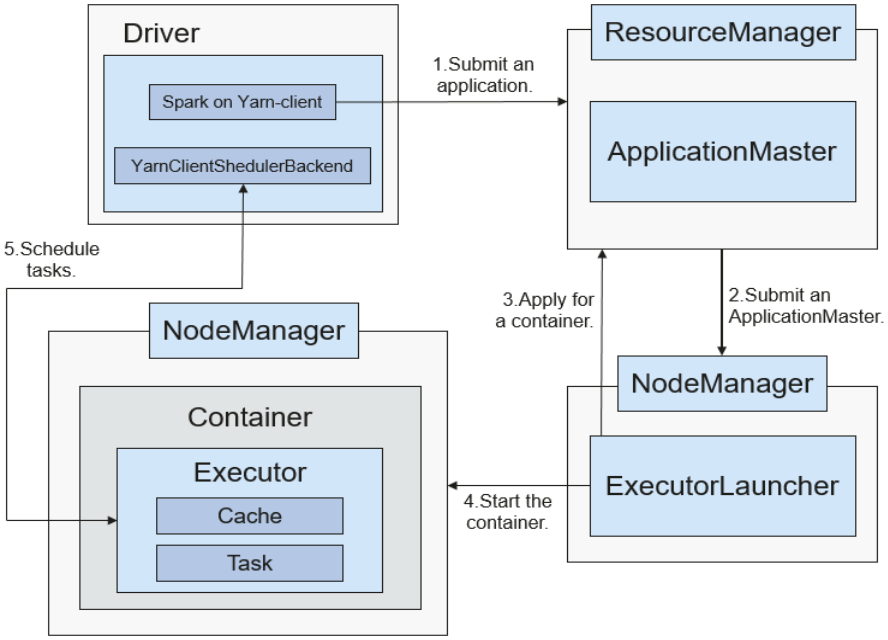
d. Drivers allocate tasks to the executors.

e. Executors run tasks and report the operating status to Drivers.

- Yarn-client mode

  Figure 1-121 describes the operation framework.

  Figure 1-121 Spark on Yarn-client operation framework



  Spark on Yarn-client implementation process:

  NOTE

  In Yarn-client mode, the Driver is deployed and started on the client. In Yarn-client mode, the client of an earlier version is incompatible. The Yarn-cluster mode is recommended.

  a. The client sends the Spark application request to ResourceManager, and packages all information required to start ApplicationMaster and sends the information to ResourceManager. ResourceManager then returns the results to the client. The results include information such as ApplicationId, and the upper limit as well as lower limit of available resources. After receiving the request, ResourceManager finds a proper node for ApplicationMaster and starts it on this node. ApplicationMaster is a role in Yarn, and the process name in Spark is ExecutorLauncher.

  b. Based on the resource requirements of each task, ApplicationMaster can apply for a series of containers to run tasks from ResourceManager.

  c. After receiving the newly allocated container list (from ResourceManager), ApplicationMaster sends information to the related NodeManagers to start the containers.

     ResourceManager allocates the containers to ApplicationMaster, which communicates with the related NodeManagers and starts the executor in the obtained container. After the executor is started, it registers with drivers and applies for tasks.

NOTE

Running Containers will not be suspended to release resources.

d. Drivers allocate tasks to the executors. Executors run tasks and report the operating status to Drivers.

1.4.24.4 Spark2x Open Source New Features

Purpose

Compared with Spark 1.5, Spark2*x* has some new open-source features. The specific features or concepts are as follows:

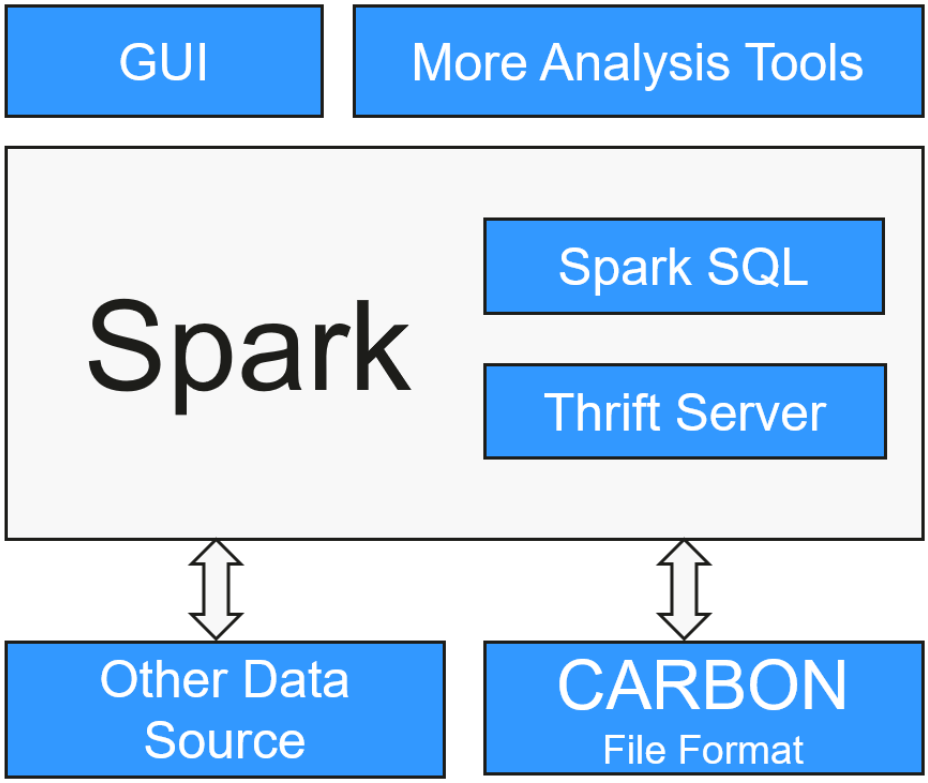
- DataSet: For details, see **SparkSQL and DataSet Principle**.
- Spark SQL Native DDL/DML: For details, see **SparkSQL and DataSet Principle**.
- SparkSession: For details, see **SparkSession Principle**.
- Structured Streaming: For details, see **Structured Streaming Principle**.
- Optimizing Small Files
- Optimizing the Aggregate Algorithm
- Optimizing Datasource Tables
- Merging CBO

1.4.24.5 Spark2x Enhanced Open Source Features

1.4.24.5.1 CarbonData Overview

CarbonData is a new Apache Hadoop native data-store format. CarbonData allows faster interactive queries over PetaBytes of data using advanced columnar storage, index, compression, and encoding techniques to improve computing efficiency. In addition, CarbonData is also a high-performance analysis engine that integrates data sources with Spark.

Figure 1-122 Basic architecture of CarbonData



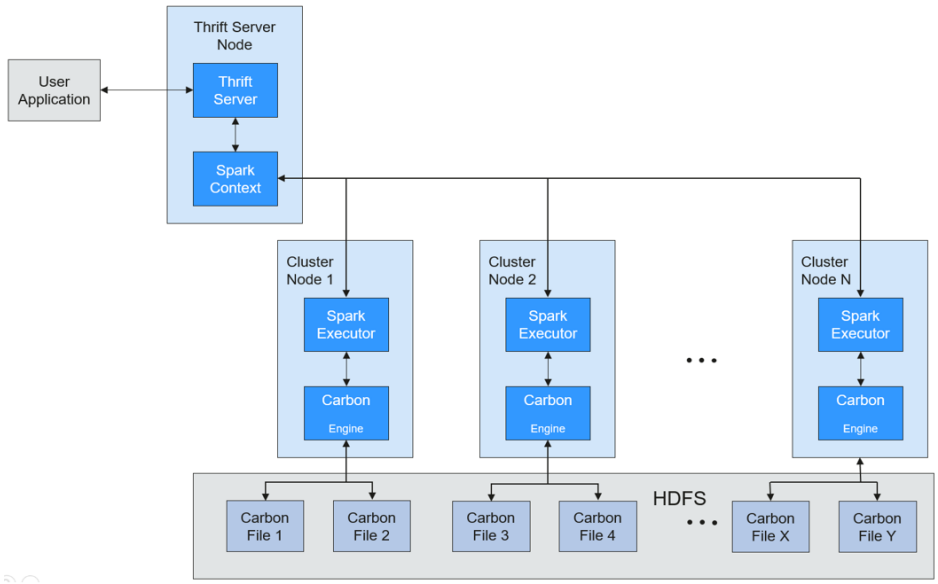
The purpose of using CarbonData is to provide quick response to ad hoc queries of big data. Essentially, CarbonData is an Online Analytical Processing (OLAP) engine, which stores data by using tables similar to those in Relational Database Management System (RDBMS). You can import more than 10 TB data to tables created in CarbonData format, and CarbonData automatically organizes and stores data using the compressed multi-dimensional indexes. After data is loaded to CarbonData, CarbonData responds to ad hoc queries in seconds.

CarbonData integrates data sources into the Spark ecosystem and you can query and analyze the data using Spark SQL. You can also use the third-party tool JDBCServer provided by Spark to connect to SparkSQL.

Topology of CarbonData

CarbonData runs as a data source inside Spark. Therefore, CarbonData does not start any additional processes on nodes in clusters. CarbonData engine runs inside the Spark executor.

Figure 1-123 Topology of CarbonData



Data stored in CarbonData Table is divided into several CarbonData data files. Each time when data is queried, CarbonData Engine reads and filters data sets. CarbonData Engine runs as a part of the Spark Executor process and is responsible for handling a subset of data file blocks.

Table data is stored in HDFS. Nodes in the same Spark cluster can be used as HDFS data nodes.

CarbonData Features

- SQL: CarbonData is compatible with Spark SQL and supports SQL query operations performed on Spark SQL.
- Simple Table dataset definition: CarbonData allows you to define and create datasets by using user-friendly Data Definition Language (DDL) statements. CarbonData DDL is flexible and easy to use, and can define complex tables.
- Easy data management: CarbonData provides various data management functions for data loading and maintenance. CarbonData supports bulk loading of historical data and incremental loading of new data. Loaded data can be deleted based on load time and a specific loading operation can be undone.
- CarbonData file format is a columnar store in HDFS. This format has many new column-based file storage features, such as table splitting and data compression. CarbonData has the following characteristics:
  - Stores data along with index: Significantly accelerates query performance and reduces the I/O scans and CPU resources, when there are filters in the query. CarbonData index consists of multiple levels of indices. A processing framework can leverage this index to reduce the task that needs to be schedules and processed, and it can also perform skip scan in more finer grain unit (called blocklet) in task side scanning instead of scanning the whole file.
  - Operable encoded data: Through supporting efficient compression and global encoding schemes, CarbonData can query on compressed/encoded

data. The data can be converted just before returning the results to the users, which is called late materialized.

- Supports various use cases with one single data format: like interactive OLAP-style query, sequential access (big scan), and random access (narrow scan).

Key Technologies and Advantages of CarbonData

- Quick query response: CarbonData features high-performance query. The query speed of CarbonData is 10 times of that of Spark SQL. It uses dedicated data formats and applies multiple index technologies, global dictionary code, and multiple push-down optimizations, providing quick response to TB-level data queries.
- Efficient data compression: CarbonData compresses data by combining the lightweight and heavyweight compression algorithms. This significantly saves 60% to 80% data storage space and the hardware storage cost.

CarbonData Index Cache Server

To solve the pressure and problems brought by the increasing data volume to the driver, an independent index cache server is introduced to separate the index from the Spark application side of Carbon query. All index content is managed by the index cache server. Spark applications obtain required index data in RPC mode. In this way, a large amount of memory on the service side is released so that services are not affected by the cluster scale and the performance or functions are not affected.

1.4.24.5.2 Optimizing SQL Query of Data of Multiple Sources

Scenario

Enterprises usually store massive data, such as from various databases and warehouses, for management and information collection. However, diversified data sources, hybrid dataset structures, and scattered data storage lower query efficiency.

The open source Spark only supports simple filter pushdown during querying of multi-source data. The SQL engine performance is deteriorated due of a large amount of unnecessary data transmission. The pushdown function is enhanced, so that **aggregate**, complex **projection**, and complex **predicate** can be pushed to data sources, reducing unnecessary data transmission and improving query performance.

Only the JDBC data source supports pushdown of query operations, such as **aggregate**, **projection**, **predicate**, **aggregate over inner join**, and **aggregate over union all**. All pushdown operations can be enabled based on your requirements.

Table 1-22 Enhanced query of cross-source query

Module	Before Enhancement	After Enhancement
aggregate	The pushdown of **aggregate** is not supported.	• Aggregation functions including **sum**, **avg**, **max**, **min**, and **count** are supported. Example: select count(*) from table • Internal expressions of aggregation functions are supported. Example: select sum(a+b) from table • Calculation of aggregation functions is supported. Example: select avg(a) + max(b) from table • Pushdown of **having** is supported. Example: select sum(a) from table where a>0 group by b having sum(a)>10 • Pushdown of some functions is supported. Pushdown of lines in mathematics, time, and string functions, such as **abs()**, **month()**, and **length()** are supported. In addition to the preceding built-in functions, you can run the **SET** command to add functions supported by data sources. Example: select sum(abs(a)) from table • Pushdown of **limit** and **order by** after **aggregate** is supported. However, the pushdown is not supported in Oracle, because Oracle does not support **limit**. Example: select sum(a) from table where a>0 group by b order by sum(a) limit 5
projection	Only pushdown of simple **projection** is supported. Example: select a, b from table	• Complex expressions can be pushed down. Example: select (a+b)*c from table • Some functions can be pushed down. For details, see the description below the table. Example: select length(a)+abs(b) from table • Pushdown of **limit** and **order by** after **projection** is supported. Example: select a, b+c from table order by a limit 3

Module	Before Enhancement	After Enhancement
predicate	Only simple filtering with the column name on the left of the operator and values on the right is supported. Example: select * from table where a>0 or b in ("aaa", "bbb")	• Complex expression pushdown is supported. Example: select * from table where a +b>c*d or a/c in (1, 2, 3) • Some functions can be pushed down. For details, see the description below the table. Example: select * from table where length(a)>5
aggregate over inner join	Related data from the two tables must be loaded to Spark. The join operation must be performed before the **aggregate** operation.	The following functions are supported: • Aggregation functions including **sum**, **avg**, **max**, **min**, and **count** are supported. • All **aggregate** operations can be performed in a same table. The **group by** operations can be performed on one or two tables and only inner join is supported. The following scenarios are not supported: • **aggregate** cannot be pushed down from both the left- and right-join tables. • **aggregate** contains operations, for example, sum(a+b). • **aggregate** operations, for example, sum(a)+min(b).
aggregate over union all	Related data from the two tables must be loaded to Spark. **union** must be performed before **aggregate**.	Supported scenarios: Aggregation functions including **sum**, **avg**, **max**, **min**, and **count** are supported. Unsupported scenarios: • **aggregate** contains operations, for example, sum(a+b). • **aggregate** operations, for example, sum(a)+min(b).

Precautions

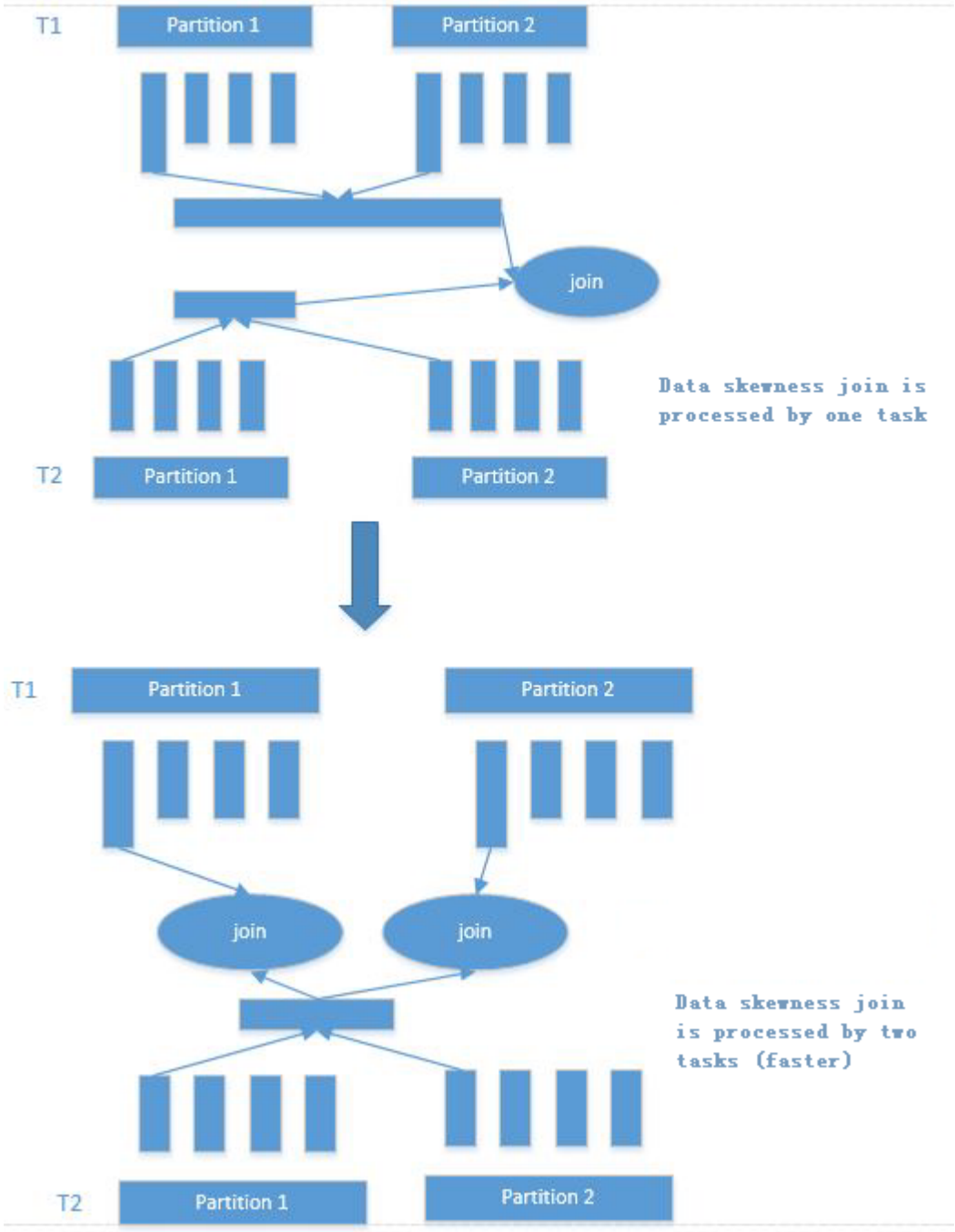
- If external data source is Hive, query operation cannot be performed on foreign tables created by Spark.
- Only MySQL and MPPDB data sources are supported.

1.4.24.5.3 Data Skewness Optimization

In the Spark SQL multi-table join scenario, severe association skewness may occur. As a result, data in some buckets is far more than that in others after data distribution by using Hash. In this case, some tasks are overloaded and run slowly while other tasks are light and run fast. Heavy tasks run slowly hindering computing performance and light tasks will result in idle CPUs, wasting CPU resources.

If there is data skewness, you are advised to configure the **spark.sql.adaptive.skewjoin.threshold** parameter to enable data skewness optimization and view data volumes of buckets. If the data volume in one bucket is too large and data skewness occurs, split the bucket and process skewed data with multiple tasks. Each task pulls full data in buckets with same join tables, improving CPU resource usage and overall performance.

Figure 1-124 Converting skewed data join



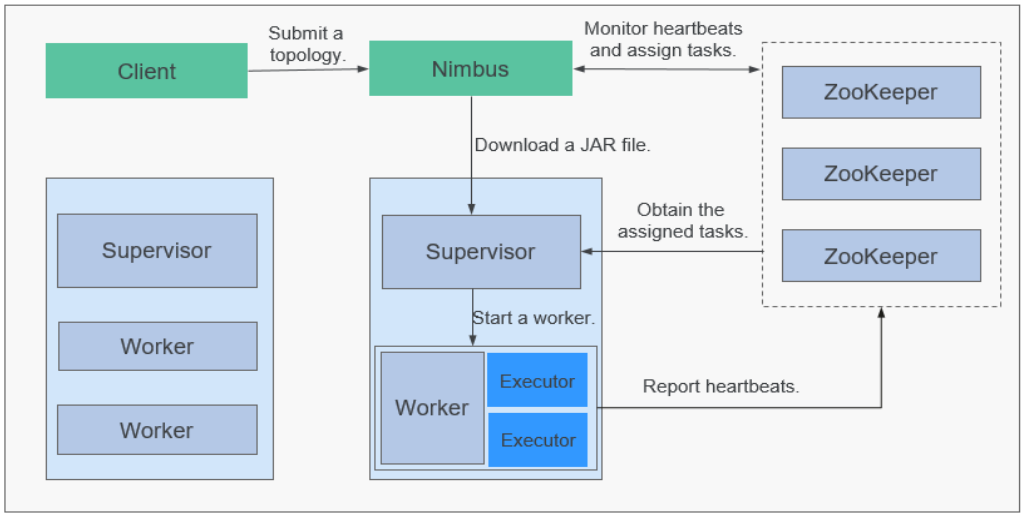
1.4.25 Storm

1.4.25.1 Storm Basic Principles

Apache Storm is a distributed, reliable, and fault-tolerant real-time stream data processing system. In Storm, a graph-shaped data structure called topology needs to be designed first for real-time computing. The topology will be submitted to a cluster. Then a master node in the cluster distributes codes and assigns tasks to

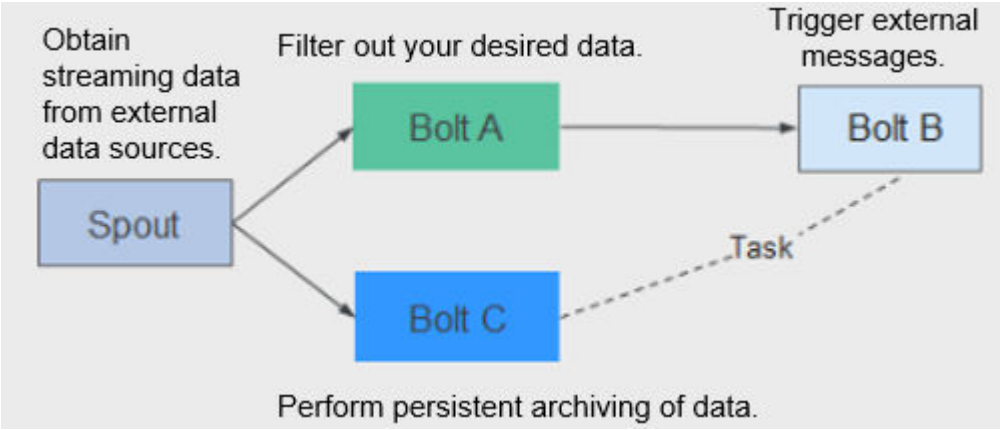
worker nodes. A topology contains two roles: spout and bolt. A spout sends messages and sends data streams in tuples. A bolt converts the data streams and performs computing and filtering operations. The bolt can randomly send data to other bolts. Tuples sent by a spout are unchangeable arrays and map to fixed key-value pairs.

Figure 1-125 System architecture of Storm



Service processing logic is encapsulated in the topology of Storm. A topology is a set of spout (data sources) and bolt (logical processing) components that are connected using Stream Groupings in DAG mode. All components (spout and bolt) in a topology are working in parallel. In a topology, you can specify the parallelism for each node. Then, Storm allocates tasks in the cluster for computing to improve system processing capabilities.

Figure 1-126 Topology



Storm is applicable to real-time analysis, continuous computing, and distributed extract, transform, and load (ETL). It has the following advantages:

- Wide applications
- High scalability

- Zero data loss
- High fault tolerance
- Easy to construct and control
- Multi-language support

Storm is a computing platform and provides Continuous Query Language (CQL) in the service layer to facilitate service implementation. CQL has the following features:

- Easy to use: The CQL syntax is similar to the SQL syntax. Users who have basic knowledge of SQL can easily learn CQL and use it to develop services.
- Rich functions: In addition to basic expressions provided by SQL, CQL provides functions, such as windows, filtering, and concurrency setting, for stream processing.
- Easy to scale: CQL provides an extension API to support increasingly complex service scenarios. Users can customize the input, output, serialization, and deserialization to meet specific service requirements.
- Easy to debug: CQL provides detailed explanation of error codes, facilitating users to rectify faults.

For details about Storm architecture and principles, see **https://storm.apache.org/**.

Principle

- **Basic Concepts**

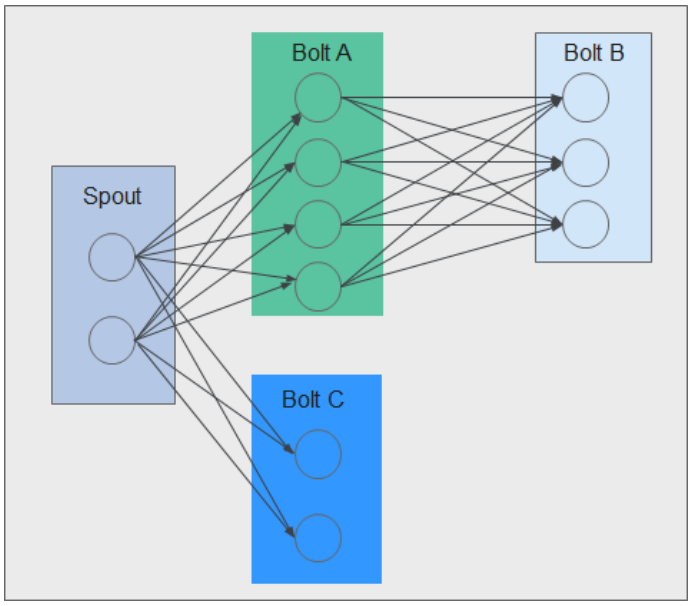
Table 1-23 Concepts

Concept	Description
Tuple	A tuple is an invariable key-value pair used to transfer data. Tuples are created and processed in distributed manner.
Stream	A stream is an unbounded sequence of tuples.
Topology	A topology is a real-time application running on the Storm platform. It is a Directed Acyclic Graph (DAG) composed of components. A topology can concurrently run on multiple machines. Each machine runs a part of the DAG. A topology is similar to a MapReduce job. The difference is that the topology is a resident program. Once started, the topology cannot stop unless it is manually terminated.
Spout	A spout is the source of tuples. For example, a spout may read data from a message queue, database, file system, or TCP connection and converts them as tuples, which are processed by the next component.

Concept	Description
Bolt	In a Topology, a bolt is a component that receives data and executes specific logic, such as filtering or converting tuples, joining or aggregating streams, and performing statistics and result persistence.
Worker	A Worker is a physical processing in running state in a Topology. Each Worker is a JVM process. Each Topology may be executed by multiple Workers. Each Worker executes a logic subset of the Topology.
Task	A task is a spout or bolt thread of a Worker.
Stream groupings	A stream grouping specifies the tuple dispatching policies. It instructs the subsequent bolt how to receive tuples. The supported policies include Shuffle Grouping, Fields Grouping, All Grouping, Global Grouping, Non Grouping, and Directed Grouping.

Figure 1-127 shows a Topology (DAG) consisting of a Spout and Bolt. In the figure, a rectangle indicates a Spout or Bolt, the node in each rectangle indicate tasks, and the lines between tasks indicate streams.

Figure 1-127 Topology



- **Reliability**

  Storm provides three levels of data reliability:

  - At Most Once: The processed data may be lost, but it cannot be processed repeatedly. This reliability level offers the highest throughput.
  - At Least Once: Data may be processed repeatedly to ensure reliable data transmission. If a response is not received within the specified time, the Spout resends the data to Bolts for processing. This reliability level may slightly affect system performance.

  - Exactly Once: Data is successfully transmitted without loss or redundancy processing. This reliability level delivers the poorest performance.

  Select the reliability level based on service requirements. For example, for the services requiring high data reliability, use Exactly Once to ensure that data is processed only once. For the services insensitive to data loss, use other levels to improve system performance.

- **Fault Tolerance**

  Storm is a fault-tolerant system that offers high availability. **Table 1-24** describes the fault tolerance of the Storm components.

  Table 1-24 Fault tolerance

Scenario	Description
Nimbus failed	Nimbus is fail-fast and stateless. If the active Nimbus is faulty, the standby Nimbus takes over services immediately, and provide external services.
Supervisor failed	Supervisor is a background daemon of Workers. It is fail-fast and stateless. If a Supervisor is faulty, the Workers running on the node are not affected but cannot receive new tasks. The OMS can detect the fault of the Supervisor and restart the processes.
Worker failed	If a Worker is faulty, the Supervisor on the Worker will restart it again. If the restart fails for multiple times, Nimbus reassigns tasks to other nodes.
Node failed	If a node is faulty, all the tasks being processed by the node time out and Nimbus will assign the tasks to another node for processing.

Open Source Features

- Distributed real-time computing

  In a Storm cluster, each machine supports the running of multiple work processes and each work process can create multiple threads. Each thread can execute multiple tasks. A task indicates concurrent data processing.

- High fault tolerance

  During message processing, if a node or a process is faulty, the message processing unit can be redeployed.

- Reliable messages

  Data processing methods including At-Least Once, At-Most Once, and Exactly Once are supported.

- Security mechanism

  Storm provides Kerberos-based authentication and pluggable authorization mechanisms, supports SSL Storm UI and Log Viewer UI, and supports security integration with other big data platform components (such as ZooKeeper and HDFS).

- Flexible topology defining and deployment

  The Flux framework is used to define and deploy service topologies. If the service DAG is changed, users only need to modify YAML domain specific language (DSL), but do not need to recompile or package service code.

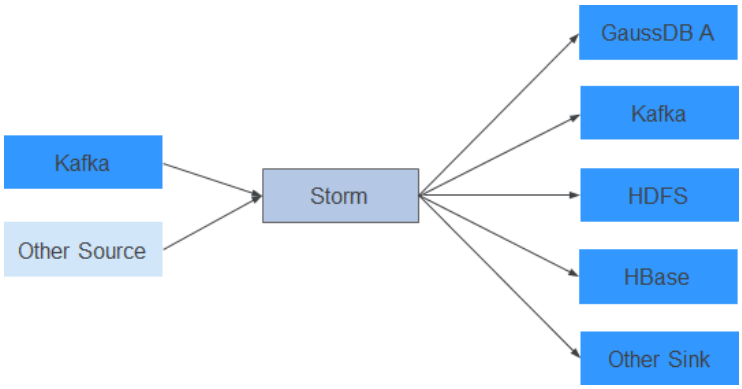
- Integration with external components

  Storm supports integration with multiple external components such as Kafka, HDFS, HBase, Redis, and JDBC/RDBMS, implementing services that involve multiple data sources.

1.4.25.2 Relationship Between Storm and Other Components

Storm provides a real-time distributed computing framework. It can obtain real-time messages from data sources (such as Kafka and TCP connection), perform high-throughput and low-latency real-time computing on a real-time platform, and export results to message queues or implement data persistence. **Figure 1-128** shows the relationship between Storm and other components.

Figure 1-128 Relationship with other components



Relationship between Storm and Streaming

Both Storm and Streaming use the open source Apache Storm kernel. However, the kernel version used by Storm is 1.2.1 whereas that used by Streaming is 0.10.0. Streaming is used to inherit transition services in upgrade scenarios. For example, if Streaming has been deployed in an earlier version and services are running, Streaming can still be used after the upgrade. Storm is recommended in a new cluster.

Storm 1.2.1 has the following new features:

- **Distributed cache**: Provides external resources (configurations) required for sharing and updating the topology using CLI tools. You do not need to re-package and re-deploy the topology.
- **Native Streaming Window API**: Provides window-based APIs.
- **Resource scheduler**: Added the resource scheduler plug-in. When defining a topology, you can specify the maximum resources available and assign resource quotas to users, thus to manage topology resources of the users.
- **State management**: Provides the Bolt API with the checkpoint mechanism. When an event fails, Storm automatically manages the Bolt status and restore the event.

- **Message sampling and debugging**: On the Storm UI, you can enable or disable topology- or component-level debugging to output stream messages to specified logs based on the sampling ratio.
- **Worker dynamic analysis**: On the Storm UI, you can collect jstack and heap logs of the Worker process and restart the Worker process.
- **Dynamic adjustment of topology logs**: You can dynamically change the running topology logs on the CLI or Storm UI.
- **Improved performance**: Compared with earlier versions, the performance of Storm is greatly improved. Although the topology performance is closely related to the use case scenario and dependency on external services, the performance is three times higher in most scenarios.

1.4.25.3 Storm Enhanced Open Source Features

- CQL

  Continuous Query Language (CQL) is an SQL-like language used for real-time stream processing. Compared with SQL, CQL has introduced the concept of (time-sequencing) window, which allows data to be stored and processed in the memory. The CQL output is the computing results of data streams at specific time. The use of CQL accelerates service development, enables tasks to be easily submitted to the Storm platform for real-time processing, facilitates output of results, and allows tasks to be terminated at the appropriate time.
- High Availability

  Nimbus HA ensures continuous service processing such as adding topologies and management even if one Nimbus is faulty, improving cluster availability.

1.4.26 Tez

Tez is Apache's latest open source computing framework that supports Directed Acyclic Graph (DAG) jobs. It can convert multiple dependent jobs into one job, greatly improving the performance of DAG jobs. If projects like Hive use Tez instead of MapReduce as the backbone of data processing, response time will be significantly reduced. Tez is built on YARN and can run MapReduce jobs without any modification.

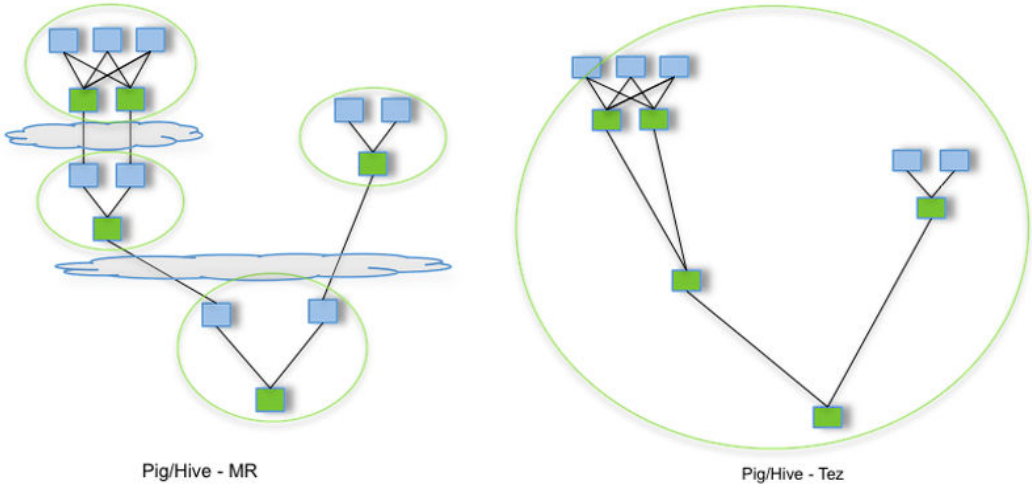
MRS uses Tez as the default execution engine of Hive. Tez remarkably surpasses the original MapReduce computing engine in terms of execution efficiency.

For details about Tez, see **https://tez.apache.org/**.

Relationship Between Tez and MapReduce

Tez uses a DAG to organize MapReduce tasks. In the DAG, a node is an RDD, and an edge indicates an operation on the RDD. The core idea is to further split Map tasks and Reduce tasks. A Map task is split into the Input-Processor-Sort-Merge-Output tasks, and the Reduce task is split into the Input-Shuffle-Sort-Merge-Process-output tasks. Tez flexibly regroups several small tasks to form a large DAG job.

Figure 1-129 Processes for submitting tasks using Hive on MapReduce and Hive on Tez



A Hive on MapReduce task contains multiple MapReduce tasks. Each task stores intermediate results to HDFS. The reducer in the previous step provides data for the mapper in the next step. A Hive on Tez task can complete the same processing process in only one task, and HDFS does not need to be accessed between tasks.

Relationship Between Tez and Yarn

Tez is a computing framework running on Yarn. The runtime environment consists of ResourceManager and ApplicationMaster of Yarn. ResourceManager is a brand new resource manager system, and ApplicationMaster is responsible for cutting MapReduce job data, assigning tasks, applying for resources, scheduling tasks, and tolerating faults. In addition, TezUI depends on TimelineServer provided by Yarn to display the running process of Tez tasks.

1.4.27 Yarn

1.4.27.1 Yarn Basic Principles

The Apache open source community introduces the unified resource management framework **Yarn** to share Hadoop clusters, improve their scalability and reliability, and eliminate a performance bottleneck of JobTracker in the early MapReduce framework.

The fundamental idea of Yarn is to split up the two major functionalities of the JobTracker, resource management and job scheduling/monitoring, into separate daemons. The idea is to have a global ResourceManager (RM) and per-application ApplicationMaster (AM).

NOTE

An application is either a single job in the classical sense of MapReduce jobs or a Directed Acyclic Graph (DAG) of jobs.

Architecture

ResourceManager is the essence of the layered structure of Yarn. This entity controls an entire cluster and manages the allocation of applications to underlying compute resources. The ResourceManager carefully allocates various resources (compute, memory, bandwidth, and so on) to underlying NodeManagers (Yarn's per-node agents). The ResourceManager also works with ApplicationMasters to allocate resources, and works with the NodeManagers to start and monitor their underlying applications. In this context, the ApplicationMaster has taken some of the role of the prior TaskTracker, and the ResourceManager has taken the role of the JobTracker.

ApplicationMaster manages each instance of an application running in Yarn. The ApplicationMaster negotiates resources from the ResourceManager and works with the NodeManagers to monitor container execution and resource usage (CPU and memory resource allocation).

The NodeManager manages each node in a Yarn cluster. The NodeManager provides per-node services in a cluster, from overseeing the management of a container over its lifecycle to monitoring resources and tracking the health of its nodes. MRv1 manages execution of the Map and Reduce tasks through slots, whereas the NodeManager manages abstract containers, which represent per-node resources available for a particular application.

Figure 1-130 Architecture

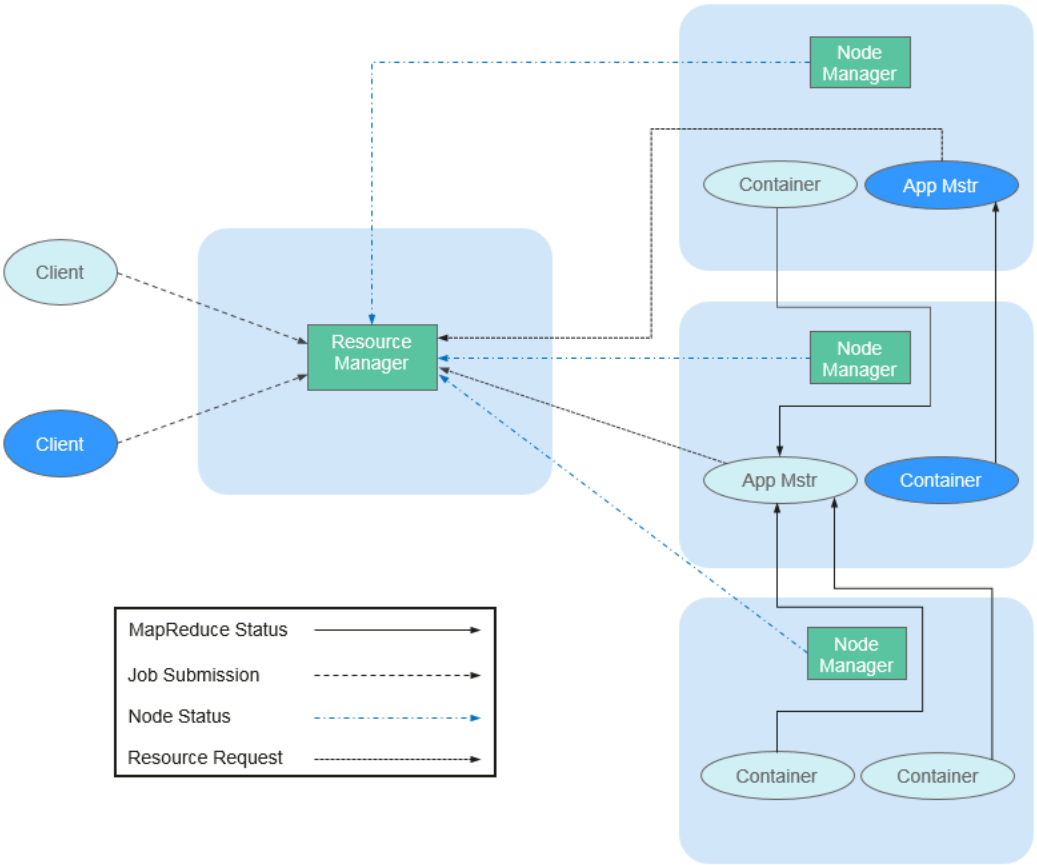


Table 1-25 describes the components shown in **Figure 1-130**.

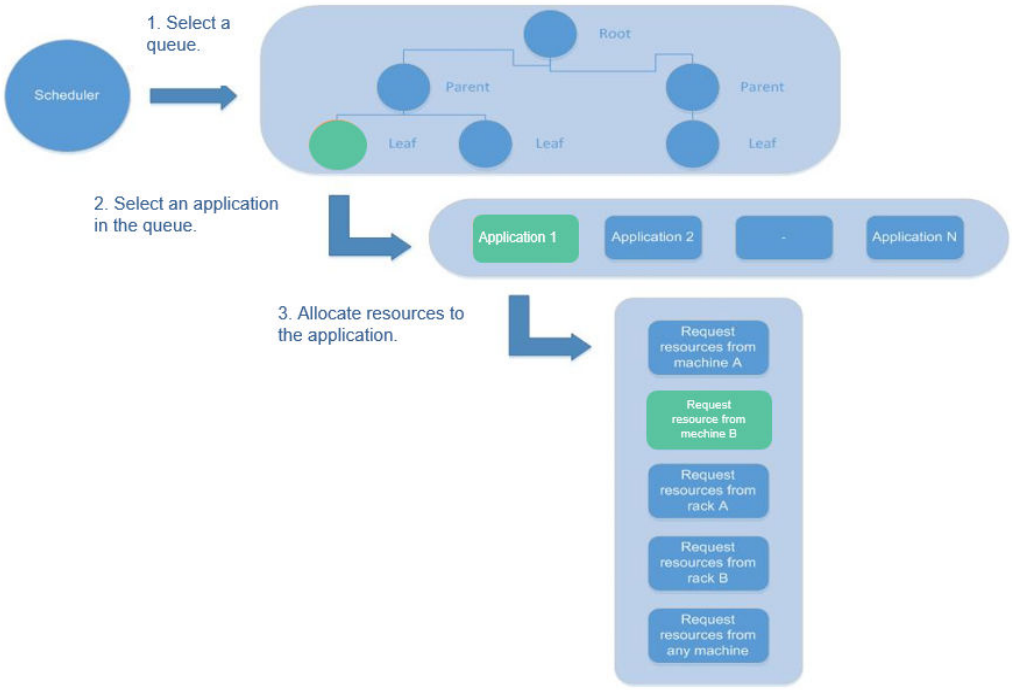
Table 1-25 Architecture description

Name	Description
Client	Client of a Yarn application. You can submit a task to ResourceManager and query the operating status of an application using the client.
ResourceManager(RM)	RM centrally manages and allocates all resources in the cluster. It receives resource reporting information from each node (NodeManager) and allocates resources to applications on the basis of the collected resources according a specified policy.
NodeManager(NM)	NM is the agent on each node of Yarn. It manages the computing node in Hadoop cluster, establishes communication with ResourceManger, monitors the lifecycle of containers, monitors the usage of resources such as memory and CPU of each container, traces node health status, and manages logs and auxiliary services used by different applications.
ApplicationMaster(AM)	AM (App Mstr in the figure above) is responsible for all tasks through the lifcycle of in an application. The tasks include the following: Negotiate with an RM scheduler to obtain a resource; further allocate the obtained resources to internal tasks (secondary allocation of resources); communicate with the NM to start or stop tasks; monitor the running status of all tasks; and apply for resources for tasks again to restart the tasks when the tasks fail to be executed.
Container	A resource abstraction in Yarn. It encapsulates multi-dimensional resources (including only memory and CPU) on a certain node. When ApplicationMaster applies for resources from ResourceManager, the ResourceManager returns resources to the ApplicationMaster in a container. Yarn allocates one container for each task and the task can only use the resources encapsulated in the container.

In Yarn, resource schedulers organize resources through hierarchical queues. This ensures that resources are allocated and shared among queues, thereby improving the usage of cluster resources. The core resource allocation model of Superior Scheduler is the same as that of Capacity Scheduler, as shown in the following figure.

A scheduler maintains queue information. You can submit applications to one or more queues. During each NM heartbeat, the scheduler selects a queue according to a specific scheduling rule, selects an application in the queue, and then allocates resources to the application. If resources fail to be allocated to the application due to the limit of some parameters, the scheduler will select another application. After the selection, the scheduler processes the resource request of this application. The scheduler gives priority to the requests for local resources first, and then for resources on the same rack, and finally for resources from any machine.

Figure 1-131 Resource allocation model



Principle

The new Hadoop MapReduce framework is named MRv2 or Yarn. Yarn consists of ResourceManager, ApplicationMaster, and NodeManager.

- ResourceManager is a global resource manager that manages and allocates resources in the system. ResourceManager consists of Scheduler and Applications Manager.
  - Scheduler allocates system resources to all running applications based on the restrictions such as capacity and queue (for example, allocates a certain amount of resources for a queue and executes a specific number of jobs). It allocates resources based on the demand of applications, with container being used as the resource allocation unit. Functioning as a dynamic resource allocation unit, Container encapsulates memory, CPU, disk, and network resources, thereby limiting the resource consumed by each task. In addition, the Scheduler is a pluggable component. You can design new schedulers as required. Yarn provides multiple directly available schedulers, such as Fair Scheduler and Capacity Scheduler.
  - Applications Manager manages all applications in the system and involves submitting applications, negotiating with schedulers about resources, enabling and monitoring ApplicationMaster, and restarting ApplicationMaster upon the startup failure.
- NodeManager is the resource and task manager of each node. On one hand, NodeManager periodically reports resource usage of the local node and the running status of each Container to ResourceManager. On the other hand, NodeManager receives and processes requests from ApplicationMaster for starting or stopping Containers.
- ApplicationMaster is responsible for all tasks through the lifecycle of an application, these channels include the following:

  - Negotiate with the RM scheduler to obtain resources.
  - Assign resources to internal components (secondary allocation of resources).
  - Communicates with NodeManager to start or stop tasks.
  - Monitor the running status of all tasks, and applies for resources again for tasks when tasks fail to run to restart the tasks.

Capacity Scheduler Principle

Capacity Scheduler is a multi-user scheduler. It allocates resources by queue and sets the minimum/maximum resources that can be used for each queue. In addition, the upper limit of resource usage is set for each user to prevent resource abuse. Remaining resources of a queue can be temporarily shared with other queues.

Capacity Scheduler supports multiple queues. It configures a certain amount of resources for each queue and adopts the first-in-first-out queuing (FIFO) scheduling policy. To prevent one user's applications from exclusively using the resources in a queue, Capacity Scheduler sets a limit on the number of resources used by jobs submitted by one user. During scheduling, Capacity Scheduler first calculates the number of resources required for each queue, and selects the queue that requires the least resources. Then, it allocates resources based on the job priority and time that jobs are submitted as well as the limit on resources and memory. Capacity Scheduler supports the following features:

- Guaranteed capacity: As the MRS cluster administrator, you can set the lower and upper limits of resource usage for each queue. All applications submitted to this queue share the resources.
- High flexibility: Temporarily, the remaining resources of a queue can be shared with other queues. However, such resources must be released in case of new application submission to the queue. Such flexible resource allocation helps notably improve resource usage.
- Multi-tenancy: Multiple users can share a cluster, and multiple applications can run concurrently. To avoid exclusive resource usage by a single application, user, or queue, the MRS cluster administrator can add multiple constraints (for example, limit on concurrent tasks of a single application).
- Assured protection: An ACL list is provided for each queue to strictly limit user access. You can specify the users who can view your application status or control the applications. Additionally, the administrator can specify a queue administrator and a cluster system administrator.
- Dynamic update of configuration files: The MRS cluster administrators can dynamically modify configuration parameters to manage clusters online.

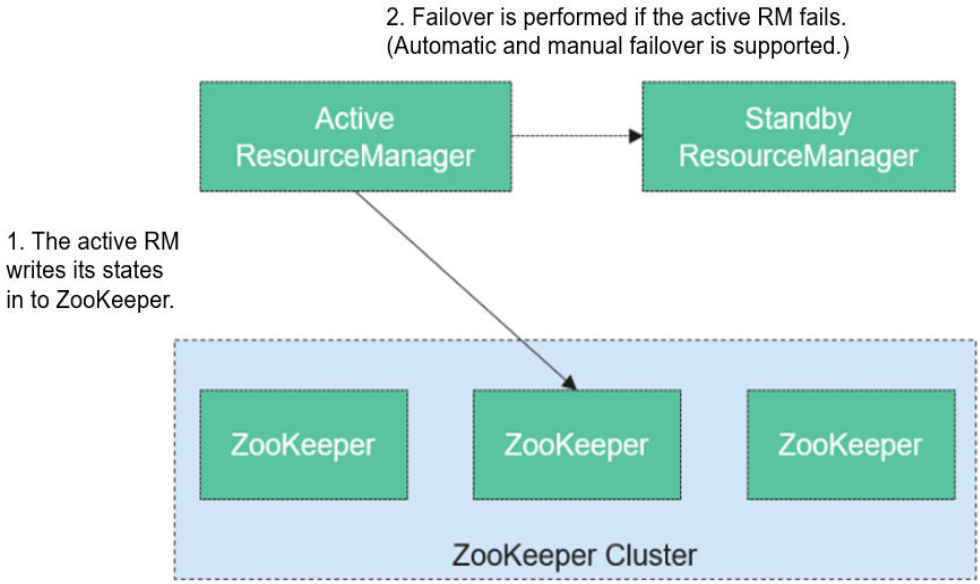
Each queue in Capacity Scheduler can limit the resource usage. However, the resource usage of a queue determines its priority when resources are allocated to queues, indicating that queues with smaller capacity are competitive. If the throughput of a cluster is big, delay scheduling enables an application to give up cross-machine or cross-rack scheduling, and to request local scheduling.

1.4.27.2 Yarn HA Solution

HA Principles and Implementation Solution

ResourceManager in Yarn manages resources and schedules tasks in the cluster. In versions earlier than Hadoop 2.4, SPOFs may occur on ResourceManager in the Yarn cluster. The Yarn HA solution uses redundant ResourceManager nodes to tackle challenges of service reliability and fault tolerance.

Figure 1-132 ResourceManager HA architecture



ResourceManager HA is achieved using active-standby ResourceManager nodes, as shown in **Figure 1-132**. Similar to the HDFS HA solution, the ResourceManager HA allows only one ResourceManager node to be in the active state at any time. When the active ResourceManager fails, the active-standby switchover can be triggered automatically or manually.

When the automatic failover function is not enabled, after the Yarn cluster is enabled, the MRS cluster administrators need to run the ***yarn rmadmin*** command to manually switch one of the ResourceManager nodes to the active state. Upon a planned maintenance event or a fault, they are expected to first demote the active ResourceManager to the standby state and the standby ResourceManager promote to the active state.

When the automatic switchover is enabled, a built-in ActiveStandbyElector that is based on ZooKeeper decide which ResourceManager node should be the active one. When the active ResourceManager is faulty, another ResourceManager node is automatically selected to be the active one to take over the faulty node.

When ResourceManager nodes in the cluster are deployed in HA mode, the configuration **yarn-site.xml** used by clients needs to list all the ResourceManager nodes. The client (including ApplicationMaster and NodeManager) searches for the active ResourceManager in polling mode. That is, the client needs to provide the fault tolerance mechanism. If the active ResourceManager cannot be connected with, the client continuously searches for a new one in polling mode.

After the standby ResourceManager promotes to be the active one, the upper-layer applications can recover to their status when the fault occurs. (For details, see **ResourceManger Restart**.) When ResourceManager Restart is enabled, the restarted ResourceManager node loads the information of the previous active ResourceManager node, and takes over container status information on all NodeManager nodes to continue service running. In this way, status information can be saved by periodically executing checkpoint operations, avoiding data loss. Ensure that both active and standby ResourceManager nodes can access the status information. Currently, three methods are provided for sharing status information by file system (FileSystemRMStateStore), LevelDB database (LeveldbRMStateStore), and ZooKeeper (ZKRMStateStore). Among them, only ZKRMStateStore supports the Fencing mechanism. By default, Hadoop uses ZKRMStateStore.

1.4.27.3 Relationship Between YARN and Other Components

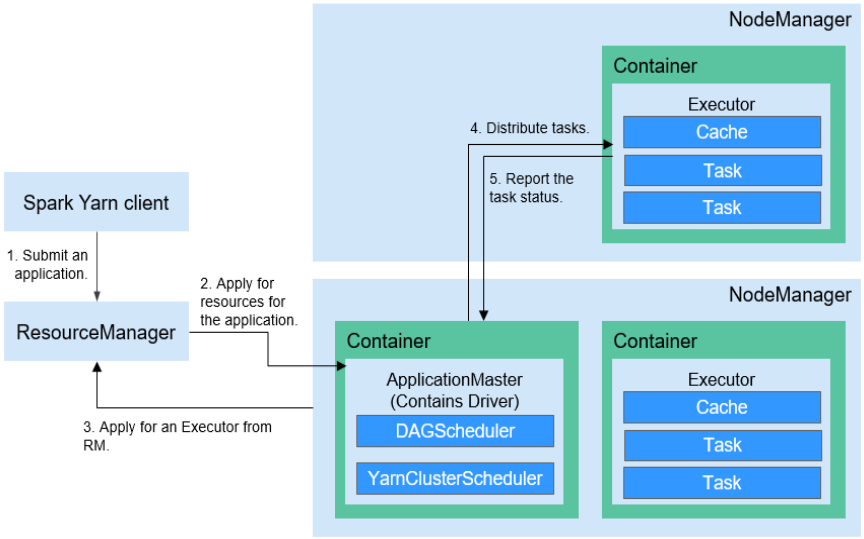
Relationship Between YARN and Spark

The Spark computing and scheduling can be implemented using YARN mode. Spark enjoys the compute resources provided by YARN clusters and runs tasks in a distributed way. Spark on YARN has two modes: YARN-cluster and YARN-client.

- YARN Cluster mode

  Figure 1-133 describes the operation framework.

  Figure 1-133 Spark on YARN-cluster operation framework



  Spark on YARN-cluster implementation process:

  a. The client generates the application information, and then sends the information to ResourceManager.

  b. ResourceManager allocates the first container (ApplicationMaster) to SparkApplication and starts the driver on the container.

  c. ApplicationMaster applies for resources from ResourceManager to run the container.

ResourceManager allocates the containers to ApplicationMaster, which communicates with the related NodeManagers and starts the executor in the obtained container. After the executor is started, it registers with drivers and applies for tasks.

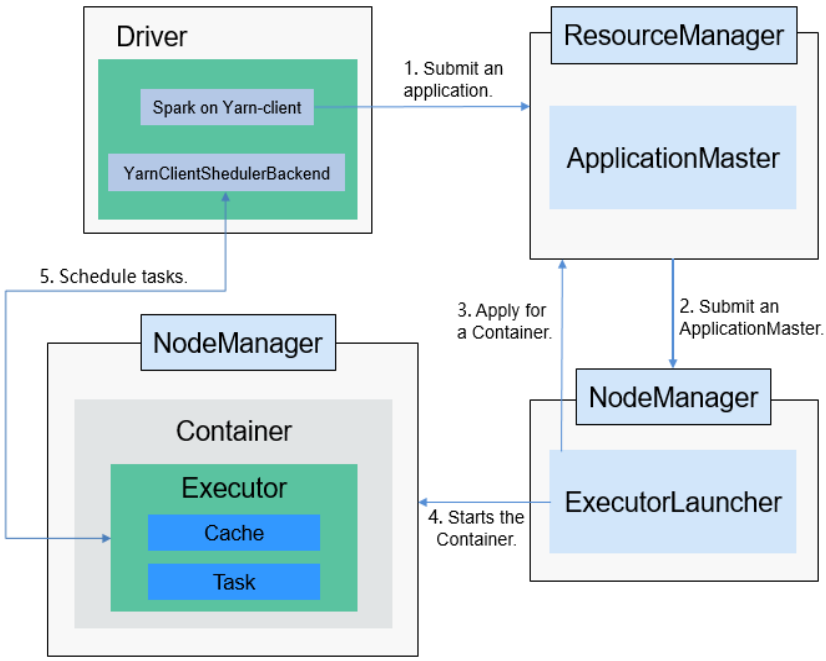
d. Drivers allocate tasks to the executors.

e. Executors run tasks and report the operating status to Drivers.

- YARN Client mode

  Figure 1-134 describes the operation framework.

  Figure 1-134 Spark on YARN-client operation framework



Spark on YARN-client implementation process:

NOTE

In YARN-client mode, the driver is deployed and started on the client. In YARN-client mode, the client of an earlier version is incompatible. You are advised to use the YARN-cluster mode.

a. The client sends the Spark application request to ResourceManager, then ResourceManager returns the results. The results include information such as Application ID and the maximum and minimum available resources. The client packages all information required to start ApplicationMaster, and sends the information to ResourceManager.

b. After receiving the request, ResourceManager finds a proper node for ApplicationMaster and starts it on this node. ApplicationMaster is a role in YARN, and the process name in Spark is ExecutorLauncher.

c. Based on the resource requirements of each task, ApplicationMaster can apply for a series of containers to run tasks from ResourceManager.

d. After receiving the newly allocated container list (from ResourceManager), ApplicationMaster sends information to the related NodeManagers to start the containers.

ResourceManager allocates the containers to ApplicationMaster, which communicates with the related NodeManagers and starts the executor in the obtained container. After the executor is started, it registers with drivers and applies for tasks.

NOTE

Running containers are not suspended and resources are not released.

e. Drivers allocate tasks to the executors. Executors run tasks and report the operating status to Drivers.

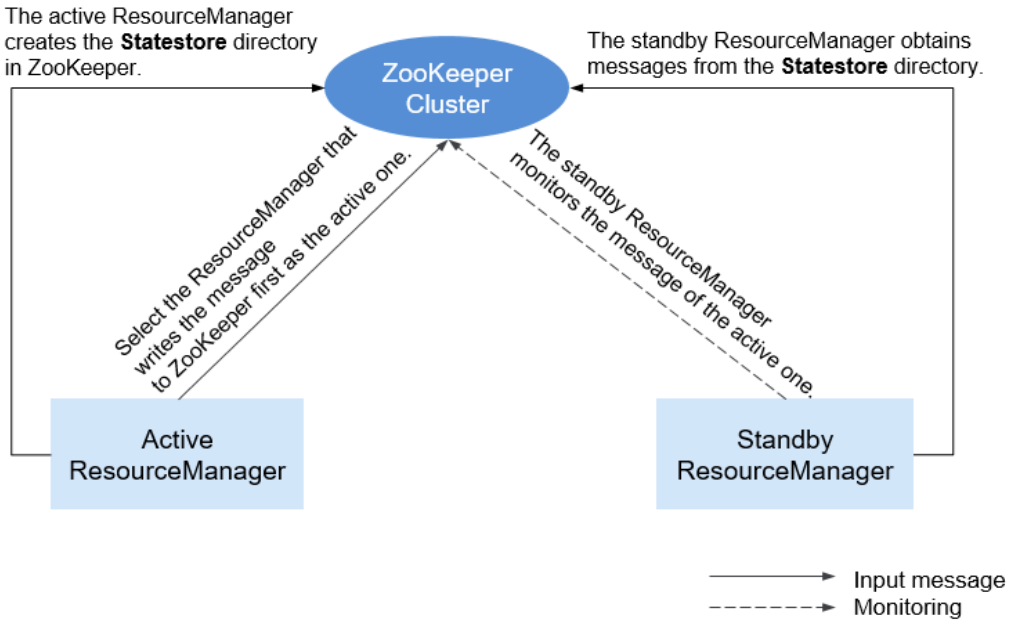
Relationship Between YARN and MapReduce

MapReduce is a computing framework running on YARN, which is used for batch processing. MRv1 is implemented based on MapReduce in Hadoop 1.0, which is composed of programming models (new and old programming APIs), running environment (JobTracker and TaskTracker), and data processing engine (MapTask and ReduceTask). This framework is still weak in scalability, fault tolerance (JobTracker SPOF), and compatibility with multiple frameworks. (Currently, only the MapReduce computing framework is supported.) MRv2 is implemented based on MapReduce in Hadoop 2.0. The source code reuses MRv1 programming models and data processing engine implementation, and the running environment is composed of ResourceManager and ApplicationMaster. ResourceManager is a brand new resource manager system, and ApplicationMaster is responsible for cutting MapReduce job data, assigning tasks, applying for resources, scheduling tasks, and tolerating faults.

Relationship Between YARN and ZooKeeper

Figure 1-135 shows the relationship between ZooKeeper and YARN.

Figure 1-135 Relationship Between ZooKeeper and YARN



1. When the system is started, ResourceManager attempts to write state information to ZooKeeper. ResourceManager that first writes state information to ZooKeeper is selected as the active ResourceManager, and others are standby ResourceManagers. The standby ResourceManagers periodically monitor active ResourceManager election information in ZooKeeper.
2. The active ResourceManager creates the **Statestore** directory in ZooKeeper to store application information. If the active ResourceManager is faulty, the standby ResourceManager obtains application information from the **Statestore** directory and restores the data.

Relationship Between YARN and Tez

The Hive on Tez job information requires the TimeLine Server capability of YARN so that Hive tasks can display the current and historical status of applications, facilitating storage and retrieval.

1.4.27.4 Yarn Enhanced Open Source Features

Priority-based task scheduling

In the native Yarn resource scheduling mechanism, if the whole Hadoop cluster resources are occupied by those MapReduce jobs submitted earlier, jobs submitted later will be kept in pending state until all running jobs are executed and resources are released.

The MRS cluster provides the task priority scheduling mechanism. With this feature, you can define jobs of different priorities. Jobs of high priority can preempt resources released from jobs of low priority though the high-priority jobs are submitted later. The low-priority jobs that are not started will be suspended unless those jobs of high priority are completed and resources are released, then they can properly be started.

This feature enables services to control computing jobs more flexibly, thereby achieving higher cluster resource utilization.

NOTE

Container reuse is in conflict with task priority scheduling. If container reuse is enabled, resources are being occupied, and task priority scheduling does not take effect.

Yarn Permission Control

The permission mechanism of Hadoop Yarn is implemented through ACLs. The following describes how to grant different permission control to different users:

- Admin ACL

  An O&M administrator is specified for the Yarn cluster. The Admin ACL is determined by **yarn.admin.acl**. The cluster O&M administrator can access the ResourceManager web UI and operate NodeManager nodes, queues, and NodeLabel, **but cannot submit tasks**.
- Queue ACL

  To facilitate user management in the cluster, users or user groups are divided into several queues to which each user and user group belongs. Each queue

contains permissions to submit and manage applications (for example, terminate any application).

Open source functions:

Currently, Yarn supports the following roles for users:

- Cluster O&M administrator
- Queue administrator
- Common user

However, the APIs (such as the web UI, REST API, and Java API) provided by Yarn do not support role-specific permission control. Therefore, all users have the permission to access the application and cluster information, which does not meet the isolation requirements in the multi-tenant scenario.

This is an enhanced function.

In security mode, permission management is enhanced for the APIs such as web UI, REST API, and Java API provided by Yarn. Permission control can be performed based on user roles.

Role-based permissions are as follows:

- Cluster O&M administrator: performs management operations in the Yarn cluster, such as accessing the ResourceManager web UI, refreshing queues, setting NodeLabel, and performing active/standby switchover.
- Queue administrator: has the permission to modify and view queues managed by the Yarn cluster.
- Common user: has the permission to modify and view self-submitted applications in the Yarn cluster.

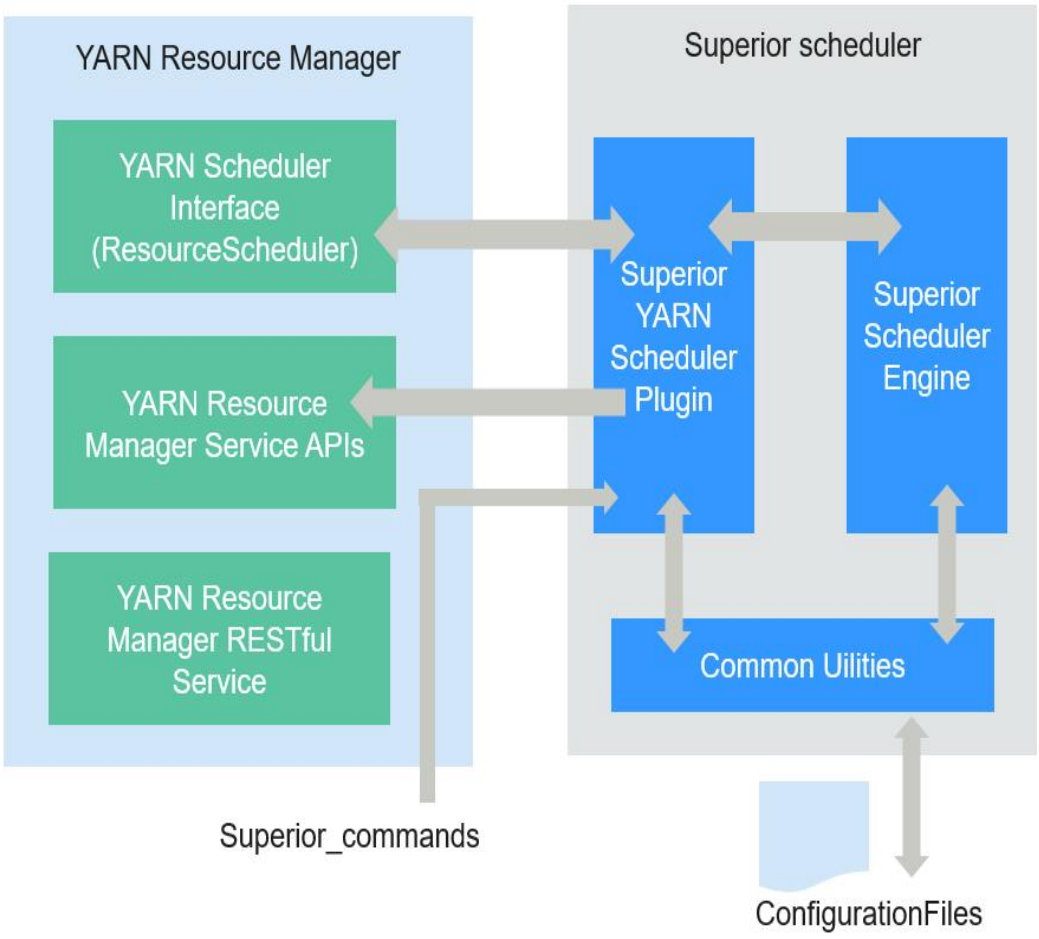
Superior Scheduler Principle (Self-developed)

Superior Scheduler is a scheduling engine designed for the Hadoop Yarn distributed resource management system. It is a high-performance and enterprise-level scheduler designed for converged resource pools and multi-tenant service requirements.

Superior Scheduler achieves all functions of open source schedulers, Fair Scheduler, and Capacity Scheduler. Compared with the open source schedulers, Superior Scheduler is enhanced in the enterprise multi-tenant resource scheduling policy, resource isolation and sharing among users in a tenant, scheduling performance, system resource usage, and cluster scalability. Superior Scheduler is designed to replace open source schedulers.

Similar to open source Fair Scheduler and Capacity Scheduler, Superior Scheduler follows the Yarn scheduler plugin API to interact with Yarn ResourceManager to offer resource scheduling functionalities. **Figure 1-136** shows the overall system diagram.

Figure 1-136 Internal architecture of Superior Scheduler



In **Figure 1-136**, Superior Scheduler consists of the following modules:

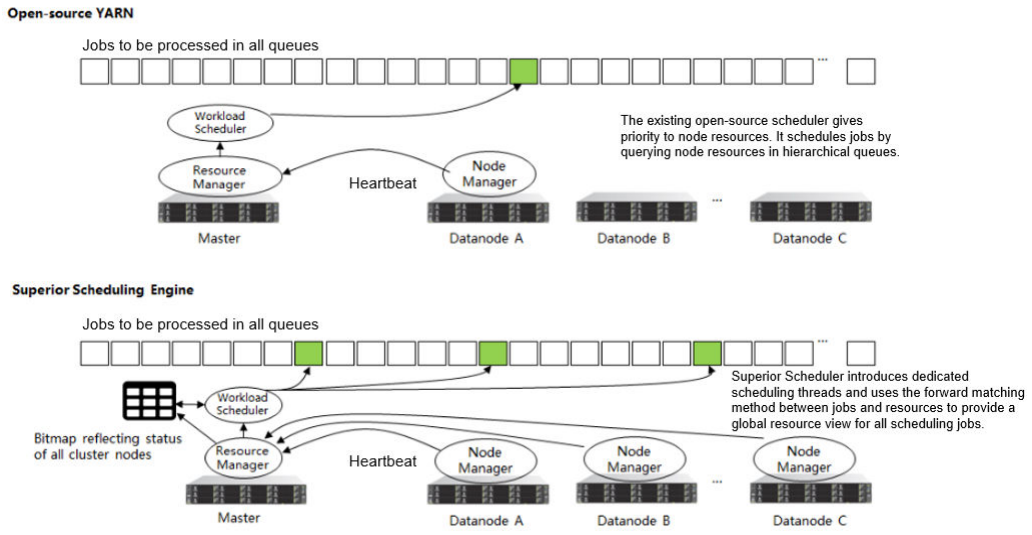
- Superior Scheduler Engine is a high performance scheduler engine with rich scheduling policies.
- Superior Yarn Scheduler Plugin functions as a bridge between Yarn ResourceManager and Superior Scheduler Engine and interacts with Yarn ResourceManager.

  The scheduling principle of open source schedulers is that resources match jobs based on the heartbeats of computing nodes. Specifically, each computing node periodically sends heartbeat messages to ResourceManager of Yarn to notify the node status and starts the scheduler to assign jobs to the node itself. In this scheduling mechanism, the scheduling period depends on the heartbeat. If the cluster scale increases, bottleneck on system scalability and scheduling performance may occur. In addition, because resources match jobs, the scheduling accuracy of an open source scheduler is limited. For example, data affinity is random and the system does not support load-based scheduling policies. The scheduler may not make the best choice due to lack of the global resource view when selecting jobs.

  Superior Scheduler adopts multiple scheduling mechanisms. There are dedicated scheduling threads in Superior Scheduler, separating heartbeats with scheduling and preventing system heartbeat storms. Additionally,

Superior Scheduler matches jobs with resources, providing each scheduled job with a global resource view and increasing the scheduling accuracy. Compared with the open source scheduler, Superior Scheduler excels in system throughput, resource usage, and data affinity.

Figure 1-137 Comparison of Superior Scheduler with open source schedulers



Apart from the enhanced system throughput and utilization, Superior Scheduler provides following major scheduling features:

- Multiple resource pools

  Multiple resource pools help logically divide cluster resources and share them among multiple tenants or queues. The division of resource pools supports heterogeneous resources. Resource pools can be divided exactly according to requirements on the application resource isolation. You can configure further policies for different queues in a pool.

- Multi-tenant scheduling (**reserve**, **min**, **share**, and **max**) in each resource pool

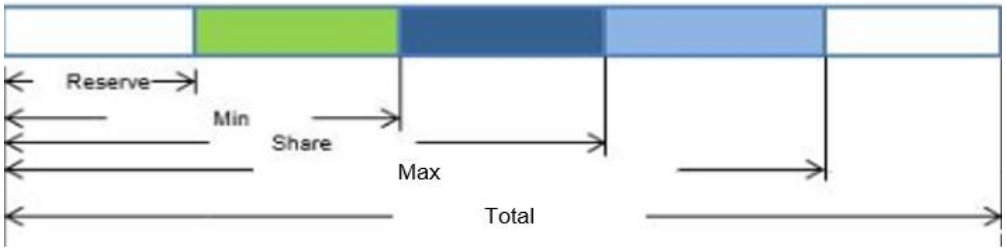
  Superior Scheduler provides flexible hierarchical multi-tenant scheduling policy. Different policies can be configured for different tenants or queues that can access different resource pools. The following figure lists supported policies:

Table 1-26 Policy description

Name	Description
reserve	This policy is used to reserve resources for a tenant. Even though tenant has no jobs available, other tenant cannot use the reserved resource. The value can be a percentage or an absolute value. If both the percentage and absolute value are configured, the percentage is automatically calculated into an absolute value, and the larger value is used. The default **reserve** value is **0**. Compared with the method of specifying a dedicated resource pool and hosts, the **reserve** policy provides a flexible floating reservation function. In addition, because no specific hosts are specified, the data affinity for calculation is improved and the impact by the faulty hosts is avoided.
min	This policy allows preemption of minimum resources. Other tenants can use these resources, but the current tenant has the priority to use them. The value can be a percentage or an absolute value. If both the percentage and absolute value are configured, the percentage is automatically calculated into an absolute value, and the larger value is used. The default value is **0**.
share	This policy is used for shared resources that cannot be preempted. To use these resources, the current tenant needs to wait for other tenants to complete jobs and release resources. The value can be a percentage or an absolute value.
max	This policy is used for the maximum resources that can be utilized. The tenant cannot obtain more resources than the allowed maximum value. The value can be a percentage or an absolute value. If both the percentage and absolute value are configured, the percentage is automatically calculated into an absolute value, and the larger value is used. By default value, there is no restriction on resources.

Figure 1-138 shows the tenant resource allocation policy.

Figure 1-138 Resource scheduling policies



NOTE

In the above figure, **Total** indicates the total number of resources, not the scheduling policy.

Compared with open source schedulers, Superior Scheduler supports both percentage and absolute value of tenants for allocating resources, flexibly addressing resource scheduling requirements of enterprise-level tenants. For example, resources can be allocated according to the absolute value of level-1 tenants, avoiding impact caused by changes of cluster scale. However, resources can be allocated according to the allocation percentage of sub-tenants, improving resource usages in the level-1 tenant.

- Heterogeneous and multi-dimensional resource scheduling

  Superior Scheduler supports following functions except CPU and memory scheduling:

  - **Node labels** can be used to identify multi-dimensional attributes of nodes such as **GPU_ENABLED** and **SSD_ENBALED**, and can be scheduled based on these labels.
  - Resource pools can be used to group resources of the same type and allocate them to specific tenants or queues.

- Fair scheduling of multiple users in a tenant

  In a leaf tenant, multiple users can use the same queue to submit jobs. Compared with the open source schedulers, Superior Scheduler supports configuring flexible resource sharing policy among different users in a same tenant. For example, VIP users can be configured with higher resource access weight.

- Data locality aware scheduling

  Superior Scheduler adopts the job-to-node scheduling policy. That is, Superior Scheduler attempts to schedule specified jobs between available nodes so that the selected node is suitable for the specified jobs. By doing so, the scheduler will have an overall view of the cluster and data. Localization is ensured if there is an opportunity to place tasks closer to the data. The open source scheduler uses the node-to-job scheduling policy to match the appropriate jobs to a given node.

- Dynamic resource reservation during container scheduling

  In a heterogeneous and diversified computing environment, some containers need more resources or multiple resources. For example, Spark job may require large memory. When such containers compete with containers requiring fewer resources, containers requiring more resources may not obtain sufficient resources within a reasonable period. Open source schedulers allocate resources to jobs, which may cause unreasonable resource reservation for these jobs. This mechanism leads to the waste of overall system resources. Superior Scheduler differs from open source schedulers in following aspects:

  - Requirement-based matching: Superior Scheduler schedules jobs to nodes and selects appropriate nodes to reserve resources to improve the startup time of containers and avoid waste.
  - Tenant rebalancing: When the reservation logic is enabled, the open source schedulers do not comply with the configured sharing policy. Superior Scheduler uses different methods. In each scheduling period, Superior Scheduler traverses all tenants and attempts to balance

resources based on the multi-tenant policy. In addition, Superior Scheduler attempts to meet all policies (**reserve**, **min**, and **share**) to release reserved resources and direct available resources to other containers that should obtain resources under different tenants.

- Dynamic queue status control (**Open**/**Closed**/**Active**/**Inactive**)

  Multiple queue statuses are supported, helping administrators operate and maintain multiple tenants.

  - Open status (**Open/Closed**): If the status is **Open** by default, applications submitted to the queue are accepted. If the status is **Closed**, no application is accepted.
  - Active status (**Active/Inactive**): If the status is **Active** by default, resources can be scheduled and allocated to applications in the tenant. Resources will not be scheduled to queues in **Inactive** status.

- Application pending reason

  If the application is not started, provide the job pending reasons.

Table 1-27 describes the comparison result of Superior Scheduler and Yarn open source schedulers.

Table 1-27 Comparative analysis

Scheduling	Yarn Open Source Scheduler	Superior Scheduler
Multi-tenant scheduling	In homogeneous clusters, either Capacity Scheduler or Fair Scheduler can be selected and the cluster does not support Fair Scheduler. Capacity Scheduler supports the scheduling by percentage and Fair Scheduler supports the scheduling by absolute value.	• Supports heterogeneous clusters and multiple resource pools. • Supports **reservation** to ensure direct access to resources.
Data locality aware scheduling	The node-to-job scheduling policy reduces the success rate of data localization and potentially affects application execution performance.	The **job-to-node scheduling policy** can aware data location more accurately, and the job hit rate of data localization scheduling is higher.
Balanced scheduling based on load of hosts	Not supported	**Balanced scheduling can be achieved when Superior Scheduler considers the host load and resource allocation during scheduling.**
Fair scheduling of multiple users in a tenant	Not supported	Supports keywords **default** and **others**.

Scheduling	Yarn Open Source Scheduler	Superior Scheduler
Job waiting reason	Not supported	Job waiting reasons illustrate why a job needs to wait.

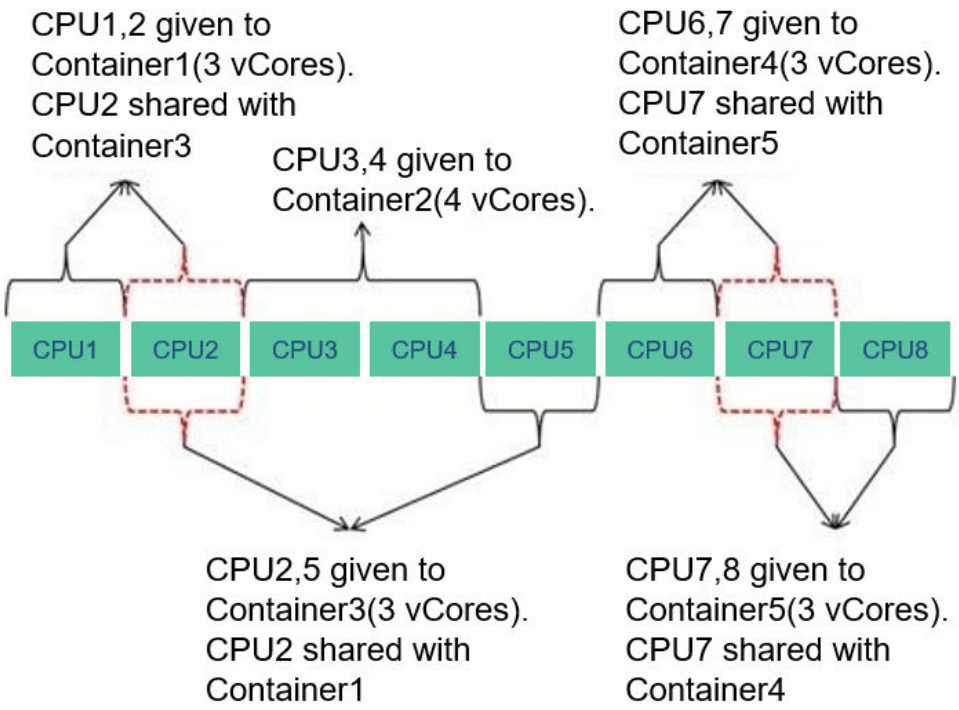
In conclusion, Superior Scheduler is a high-performance scheduler with various scheduling policies and is better than Capacity Scheduler in terms of functionality, performance, resource usage, and scalability.

CPU Hard Isolation

Yarn cannot strictly control the CPU resources used by each container. When the CPU subsystem is used, a container may occupy excessive resources. Therefore, CPUset is used to control resource allocation.

To solve this problem, the CPU resources are allocated to each container based on the ratio of virtual cores (vCores) to physical cores. If a container requires an entire physical core, the container has it. If a container needs only some physical cores, several containers may share the same physical core. The following figure shows an example of the CPU quota. The given ratio of vCores to physical cores is 2:1.

Figure 1-139 CPU quota



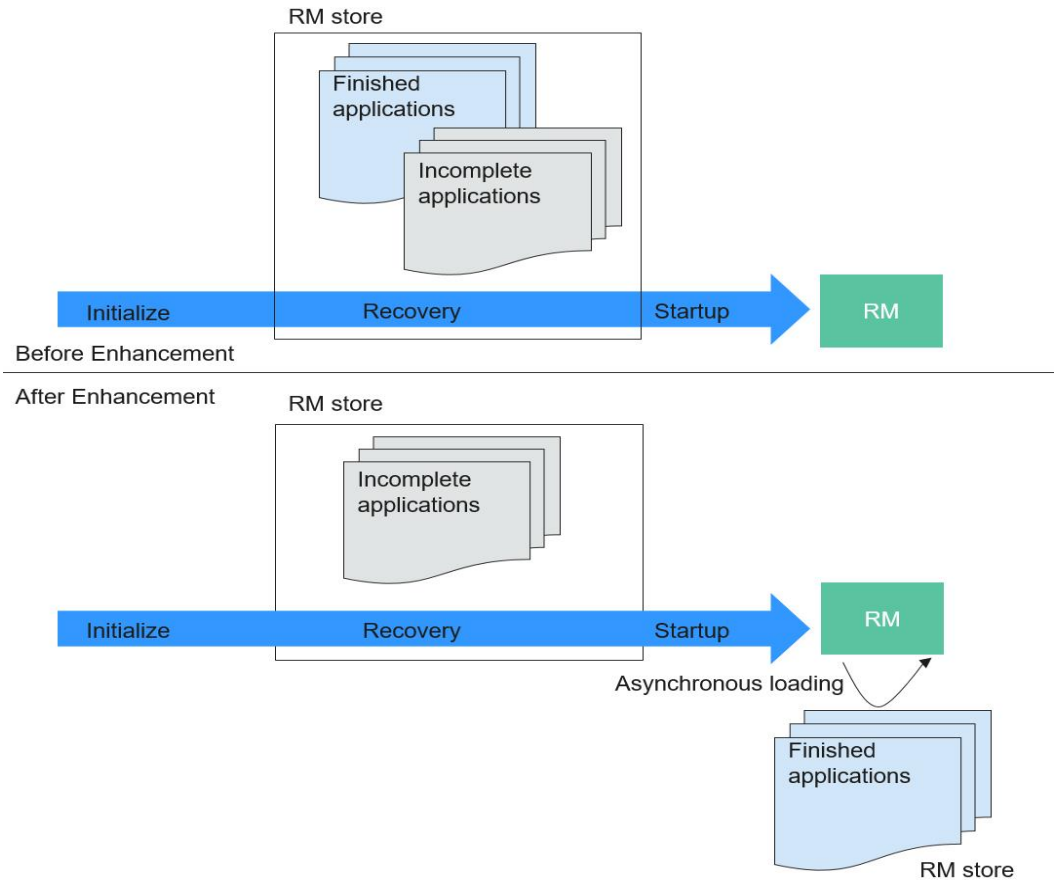
Enhanced Open Source Feature: Optimizing Restart Performance

Generally, the recovered ResourceManager can obtain running and completed applications. However, a large number of completed applications may cause

problems such as slow startup and long HA switchover/restart time of ResourceManagers.

To speed up the startup, obtain the list of unfinished applications before starting the ResourceManagers. In this case, the completed application continues to be recovered in the background asynchronous thread. The following figure shows how the ResourceManager recovery starts.

Figure 1-140 Starting the ResourceManager recovery



1.4.28 ZooKeeper

1.4.28.1 ZooKeeper Basic Principle

ZooKeeper Overview

ZooKeeper is a distributed, highly available coordination service. ZooKeeper provides two functions:

- Prevents the system from single point of failures (SPOFs) and provides reliable services for applications.
- Provides distributed coordination services and manages configuration information.

ZooKeeper Architecture

Nodes in a ZooKeeper cluster have three roles: Leader, Follower, and Observer. Figure 1-141 shows the ZooKeeper architecture. Generally, an odd number (2N+1) of ZooKeeper servers are configured. At least (N+1) vote majority is required to successfully perform write operation.

Figure 1-141 ZooKeeper architecture

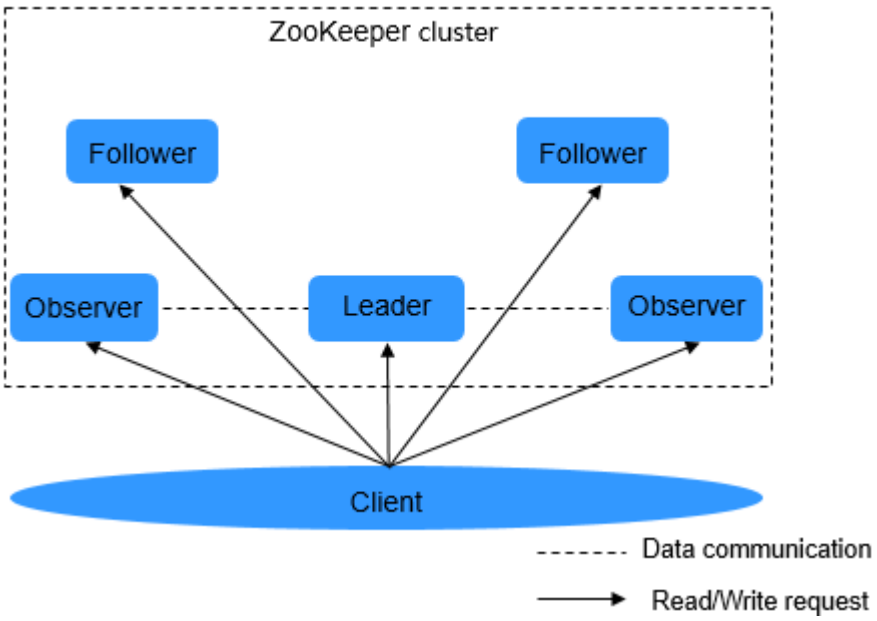


Table 1-28 describes the functions of each module shown in Figure 1-141.

Table 1-28 ZooKeeper modules

Module	Description
Leader	Only one node serves as the Leader in a ZooKeeper cluster. The Leader, elected by Followers using the ZooKeeper Atomic Broadcast (ZAB) protocol, receives and coordinates all write requests and synchronizes written information to Followers and Observers.
Follower	Follower has two functions: • Prevents SPOF. A new Leader is elected from Followers when the Leader is faulty. • Processes read requests and interacts with the Leader to process write requests.
Observer	The Observer does not take part in voting for election and write requests. It only processes read requests and forwards write requests to the Leader, increasing system processing efficiency.

Module	Description
Client	Reads and writes data from or to the ZooKeeper cluster. For example, HBase can serve as a ZooKeeper client and use the arbitration function of the ZooKeeper cluster to control the active/standby status of the HMaster.

If security services are enabled in the cluster, authentication is required during the connection to ZooKeeper. The authentication modes are as follows:

- keytab mode: Obtain a human-machine user from the MRS cluster administrator for login to the platform and authentication, and obtain the keytab file of the user.
- Ticket mode: Obtain a human-machine user from the MRS cluster administrator for subsequent secure login, enable the renewable and forwardable functions of the Kerberos service, set the ticket update interval, and restart Kerberos and related components.

  NOTE

  - The default validity period of a user password is 90 days. Therefore, the validity period of the obtained keytab file is 90 days. To prolong the validity period of the keytab file, modify the user password policy and obtain the keytab file again. For details, see the *Administrator Guide*.
  - The parameters for enabling the renewable and forwardable functions and setting the ticket update interval are on the **System** tab of the Kerberos service configuration page. The ticket update interval can be set to **kdc_renew_lifetime** or **kdc_max_renewable_life** based on the actual situation.

ZooKeeper Principle

- **Write Request**
  a. After the Follower or Observer receives a write request, the Follower or Observer sends the request to the Leader.
  b. The Leader coordinates Followers to determine whether to accept the write request by voting.
  c. If more than half of voters return a write success message, the Leader submits the write request and returns a success message. Otherwise, a failure message is returned.
  d. The Follower or Observer returns the processing results.
- **Read Request**

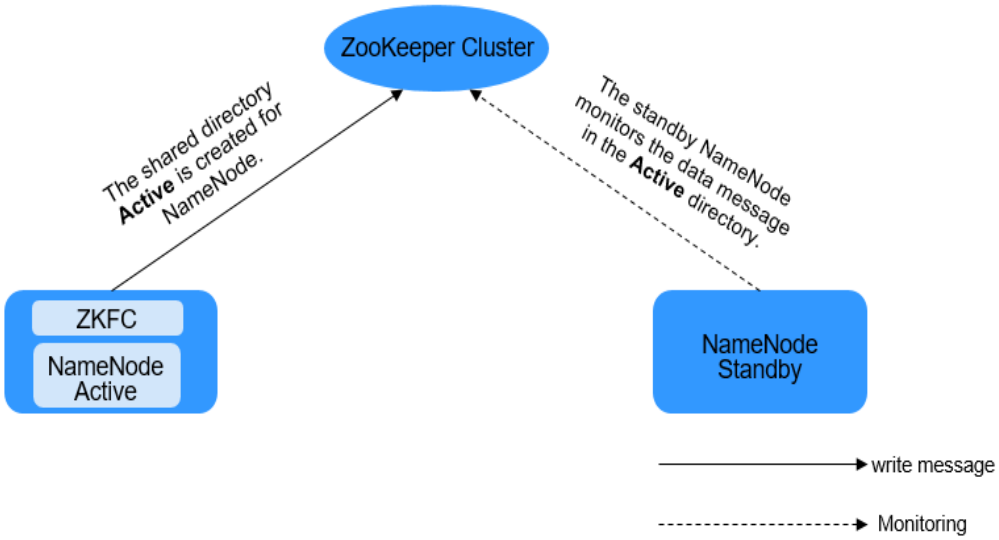
  The client directly reads data from the Leader, Follower, or Observer.

1.4.28.2 Relationship Between ZooKeeper and Other Components

Relationship Between ZooKeeper and HDFS

Figure 1-142 shows the relationship between ZooKeeper and HDFS.

Figure 1-142 Relationship between ZooKeeper and HDFS



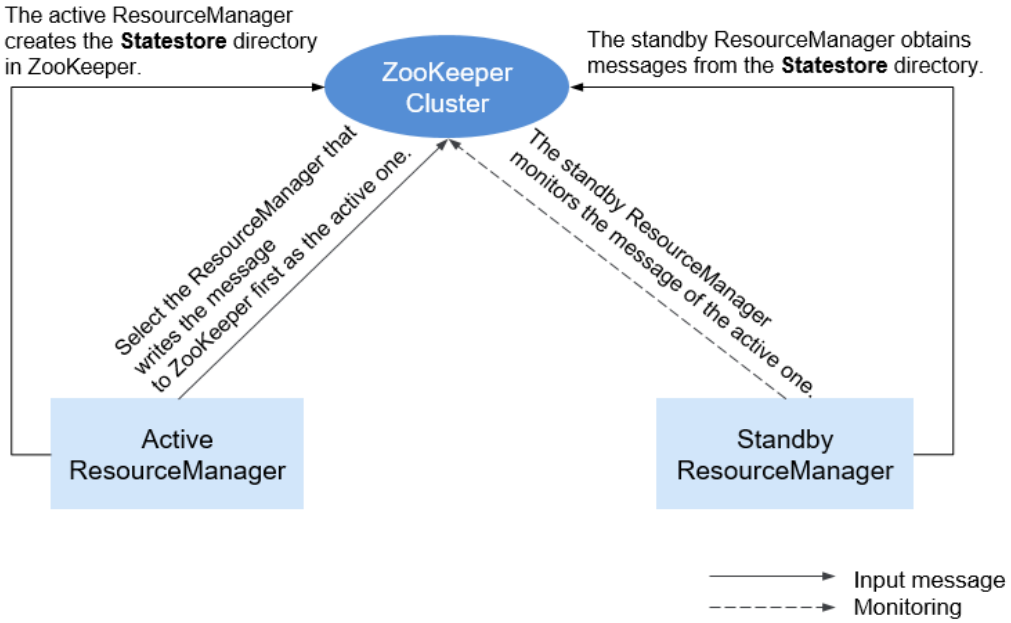
As the client of a ZooKeeper cluster, ZKFailoverController (ZKFC) monitors the status of NameNode. ZKFC is deployed only in the node where NameNode resides, and in both the active and standby HDFS NameNodes.

1. The ZKFC connects to ZooKeeper and saves information such as host names to ZooKeeper under the znode directory **/hadoop-ha**. NameNode that creates the directory first is considered as the active node, and the other is the standby node. NameNodes read the NameNode information periodically through ZooKeeper.
2. When the process of the active node ends abnormally, the standby NameNode detects changes in the **/hadoop-ha** directory through ZooKeeper, and then takes over the service of the active NameNode.

Relationship Between ZooKeeper and YARN

Figure 1-143 shows the relationship between ZooKeeper and YARN.

Figure 1-143 Relationship Between ZooKeeper and YARN

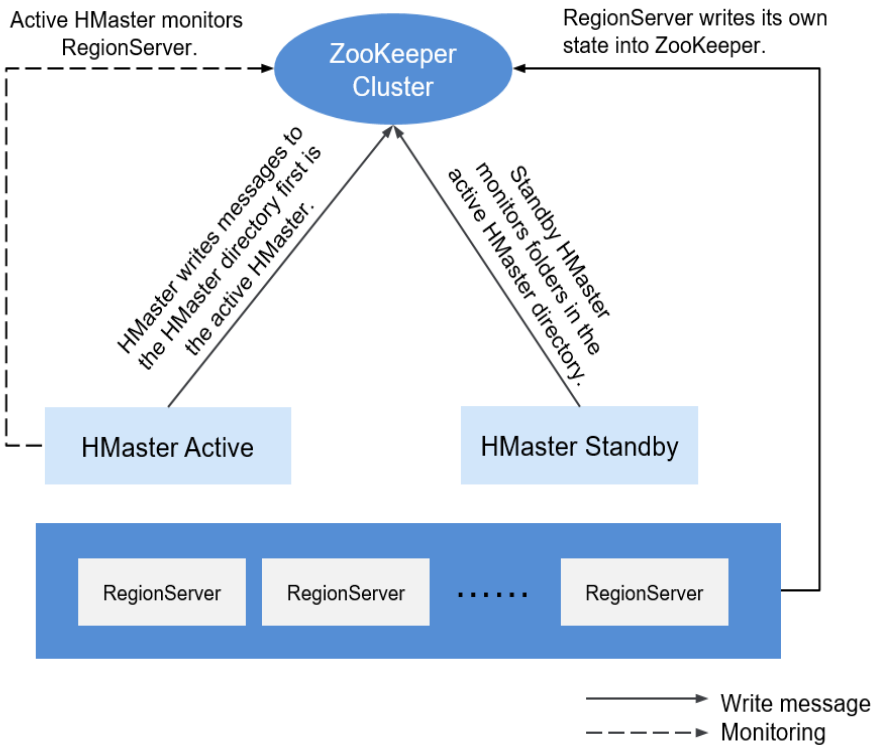


1. When the system is started, ResourceManager attempts to write state information to ZooKeeper. ResourceManager that first writes state information to ZooKeeper is selected as the active ResourceManager, and others are standby ResourceManagers. The standby ResourceManagers periodically monitor active ResourceManager election information in ZooKeeper.
2. The active ResourceManager creates the **Statestore** directory in ZooKeeper to store application information. If the active ResourceManager is faulty, the standby ResourceManager obtains application information from the **Statestore** directory and restores the data.

Relationship Between ZooKeeper and HBase

Figure 1-144 shows the relationship between ZooKeeper and HBase.

Figure 1-144 Relationship between ZooKeeper and HBase

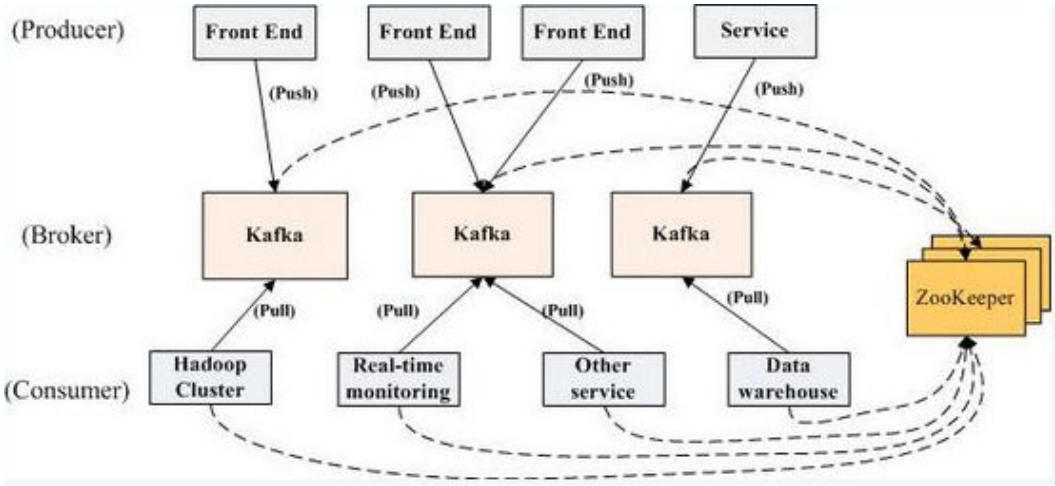


1. HRegionServer registers itself to ZooKeeper on Ephemeral node. ZooKeeper stores the HBase information, including the HBase metadata and HMaster addresses.
2. HMaster detects the health status of each HRegionServer using ZooKeeper, and monitors them.
3. HBase supports multiple HMaster nodes (like HDFS NameNodes). When the active HMatser is faulty, the standby HMaster obtains the state information about the entire cluster using ZooKeeper. That is, using ZooKeeper can avoid HBase SPOFs.

Relationship Between ZooKeeper and Kafka

Figure 1-145 shows the relationship between ZooKeeper and Kafka.

Figure 1-145 Relationship between ZooKeeper and Kafka



1. Broker uses ZooKeeper to register broker information and elect a partition leader.
2. The consumer uses ZooKeeper to register consumer information, including the partition list of consumer. In addition, ZooKeeper is used to discover the broker list, establish a socket connection with the partition leader, and obtain messages.

1.4.28.3 ZooKeeper Enhanced Open Source Features

Enhanced Log

In security mode, an ephemeral node is deleted as long as the session that created the node expires. Ephemeral node deletion is recorded in audit logs so that ephemeral node status can be obtained.

Usernames must be added to audit logs for all operations performed on ZooKeeper clients.

On the ZooKeeper client, create a znode, of which the Kerberos principal is **zkcli/hadoop.***<System domain name>***@***<System domain name>*.

For example, open the **<ZOO_LOG_DIR>/zookeeper_audit.log** file. The file content is as follows:

```
2016-12-28 14:17:10,505 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
target=ZooKeeperServer?znode=/test1?result=success
2016-12-28 14:17:10,530 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
target=ZooKeeperServer?znode=/test2?result=success
2016-12-28 14:17:10,550 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
target=ZooKeeperServer?znode=/test3?result=success
2016-12-28 14:17:10,570 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
target=ZooKeeperServer?znode=/test4?result=success
2016-12-28 14:17:10,592 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
target=ZooKeeperServer?znode=/test5?result=success
2016-12-28 14:17:10,613 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
```

```
target=ZooKeeperServer?znode=/test6?result=success
2016-12-28 14:17:10,633 | INFO  | CommitProcWorkThread-4 | session=0x12000007553b4903?
user=10.177.223.78,zkcli/hadoop.hadoop.com@HADOOP.COM?ip=10.177.223.78?operation=create znode?
target=ZooKeeperServer?znode=/test7?result=success
```

The content shows that logs of the ZooKeeper client user **zkcli/hadoop.hadoop.com@HADOOP.COM** are added to the audit log.

User details in ZooKeeper

In ZooKeeper, different authentication schemes use different credentials as users. Based on the authentication provider requirement, any parameter can be considered as users.

Example:

- **SAMLAuthenticationProvider** uses the client principal as a user.
- **X509AuthenticationProvider** uses the user client certificate as a user.
- **IAuthenticationProvider** uses the client IP address as a user.
- A username can be obtained from the custom authentication provider by implementing the **org.apache.zookeeper.server.auth.ExtAuthenticationProvider.getUserName(String)** method. If the method is not implemented, getting the username from the authentication provider instance will be skipped.

Enhanced Open Source Feature: ZooKeeper SSL Communication (Netty Connection)

The ZooKeeper design contains the Nio package and does not support SSL later than version 3.5. To solve this problem, Netty is added to ZooKeeper. Therefore, if you need to use SSL, enable Netty and set the following parameters on the server and client:

The open source server supports only plain text passwords, which may cause security problems. Therefore, such text passwords are no longer used on the server.

- Client
  a. Set **-Dzookeeper.client.secure** in the **zkCli.sh/zkEnv.sh** file to **true** to use secure communication on the client. Then, the client can connect to the secureClientPort on the server.
  b. Set the following parameters in the **zkCli.sh/zkEnv.sh** file to configure the client environment:

Parameter	Description
-Dzookeeper.clientCnxnSocket	Used for Netty communication between clients. Default value: **org.apache.zookeeper.ClientCnxnSocketNetty**
-Dzookeeper.ssl.keyStore.location	Indicates the path for storing the keystore file.

Parameter	Description
-Dzookeeper.ssl.keyStore.password	Encrypts a password.
-Dzookeeper.ssl.trustStore.location	Indicates the path for storing the truststore file.
-Dzookeeper.ssl.trustStore.password	Encrypts a password.
-Dzookeeper.config.crypt.class	Decrypts an encrypted password.
-Dzookeeper.ssl.password.encrypted	Default value: **false** If the keystore and truststore passwords are encrypted, set this parameter to **true**.
-Dzookeeper.ssl.enabled.protocols	Defines the SSL protocols to be enabled for the SSL context.
-Dzookeeper.ssl.exclude.cipher.ext	Defines the list of passwords separated by a comma which should be excluded from the SSL context.

NOTE

The preceding parameters must be set in the **zkCli.sh/zk.Env.sh** file.

- Server
  a. Set **secureClientPort** to **3381** in the **zoo.cfg** file.
  b. Set **zookeeper.serverCnxnFactory** to **org.apache.zookeeper.server.NettyServerCnxnFactory** in the **zoo.cfg** file on the server.
  c. Set the following parameters in the **zoo.cfg** file (in the **zookeeper/conf/zoo.cfg** path) to configure the server environment:

Parameter	Description
ssl.keyStore.location	Path for storing the **keystore.jks** file
ssl.keyStore.password	Encrypts a password.
ssl.trustStore.location	Indicates the path for storing the truststore file.
ssl.trustStore.password	Encrypts a password.
config.crypt.class	Decrypts an encrypted password.

Parameter	Description
ssl.keyStore.password.encrypted	Default value: **false** If this parameter is set to **true**, the encrypted password can be used.
ssl.trustStore.password.encrypted	Default value: **false** If this parameter is set to **true**, the encrypted password can be used.
ssl.enabled.protocols	Defines the SSL protocols to be enabled for the SSL context.
ssl.exclude.cipher.ext	Defines the list of passwords separated by a comma which should be excluded from the SSL context.

d. Start ZKserver and connect the security client to the security port.

- Credential

  The credential used between client and server in ZooKeeper is **X509AuthenticationProvider**. This credential is initialized using the server certificates specified and trusted by the following parameters:

  – zookeeper.ssl.keyStore.location
  – zookeeper.ssl.keyStore.password
  – zookeeper.ssl.trustStore.location
  – zookeeper.ssl.trustStore.password

  NOTE

  If you do not want to use default mechanism of ZooKeeper, then it can be configured with different trust mechanisms as needed.

1.5 Functions

1.5.1 Multi-tenant

Feature Introduction

Modern enterprises' data clusters are developing towards centralization and cloudification. Enterprise-class big data clusters must meet the following requirements:

- Carry data of different types and formats and run jobs and applications of different types (analysis, query, and stream processing).
- Isolate data of a user from that of another user who has demanding requirements on data security, such as a bank or government institute.

The preceding requirements bring the following challenges to the big data cluster:

- Proper allocation and scheduling of resources to ensure stable operating of applications and jobs
- Strict access control to ensure data and service security

Multi-tenant isolates the resources of a big data cluster into resource sets. Users can lease desired resource sets to run applications and jobs and store data. In a big data cluster, multiple resource sets can be deployed to meet diverse requirements of multiple users.

The MRS big data cluster provides a complete enterprise-class big data multi-tenant solution. Multi-tenant is a collection of multiple resources (each resource set is a tenant) in an MRS big data cluster. It can allocate and schedule resources, including computing and storage resources.

Advantages

- Proper resource configuration and isolation

  The resources of a tenant are isolated from those of another tenant. The resource use of a tenant does not affect other tenants. This mechanism ensures that each tenant can configure resources based on service requirements, improving resource utilization.
- Resource consumption measurement and statistics

  Tenants are system resource applicants and consumers. System resources are planned and allocated based on tenants. Resource consumption by tenants can be measured and recorded.
- Ensured data security and access security

  In multi-tenant scenarios, the data of each tenant is stored separately to ensure data security. The access to tenants' resources is controlled to ensure access security.

Enhanced Schedulers

Schedulers are divided into the open source Capacity scheduler and proprietary Superior scheduler.

To meet enterprise requirements and tackle challenges facing the Yarn community in scheduling, develops the Superior scheduler. In addition to inheriting the advantages of the Capacity scheduler and Fair scheduler, this scheduler is enhanced in the following aspects:

- Enhanced resource sharing policy

  The Superior scheduler supports queue hierarchy. It integrates the functions of open source schedulers and shares resources based on configurable policies. In terms of instances, the MRS cluster administrators can use the Superior scheduler to configure an absolute value or percentage policy for queue resources. The resource sharing policy of the Superior scheduler enhances the label scheduling policy of Yarn as a resource pool feature. The nodes in the Yarn cluster can be grouped based on the capacity or service type to ensure that queues can more efficiently utilize resources.
- Tenant-based resource reservation policy

  Resources required by tenants must be ensured for running critical tasks. The Superior scheduler builds a mechanism to support the resource reservation

policy. By doing so, reserved resources can be allocated to the tasks run by the tenant queues in a timely manner to ensure proper task execution.

- Fair sharing among tenants and resource pool users

  The Superior scheduler allows shared resources to be configured for users in a queue. Each tenant may have users with different weights. Heavily weighted users may require more shared resources.

- Ensured scheduling performance in a big cluster

  The Superior scheduler receives heartbeats from each NodeManager and saves resource information in memory, which enables the scheduler to control cluster resource usage globally. The Superior scheduler uses the push scheduling model, which makes the scheduling more precise and efficient and remarkably improves cluster resource utilization. Additionally, the Superior scheduler delivers excellent performance when the interval between NodeManager heartbeats is long and prevents heartbeat storms in big clusters.

- Priority policy

  If the minimum resource requirement of a service cannot be met after the service obtains all available resources, a preemption occurs. The preemption function is disabled by default.

1.5.2 Security Hardening

MRS is a platform for massive data management and analysis and has high security. MRS protects user data and service running from the following aspects:

- Network isolation

  The entire system is deployed in a VPC on the public cloud to provide an isolated network environment and ensure service and management security of the cluster. By combining the subnet division, route control, and security group functions of VPC, MRS provides a secure and reliable isolated network environment.

- Resource isolation

  MRS supports resource deployment and isolation of physical resources in dedicated zones. You can flexibly combine computing and storage resources, such as dedicated computing resources + shared storage resources, shared computing resources + dedicated storage resources, and dedicated computing resources + dedicated storage resources.

- Host security

  MRS can be integrated with public cloud security services, including Vulnerability Scan Service (VSS), Host Security Service (HSS), Web Application Firewall (WAF), Cloud Bastion Host (CBH), and Web Tamper Protection (WTP). The following measures are provided to improve security of the OS and ports:

  - Security hardening of OS kernels
  - OS patch update
  - OS permission control
  - OS port management
  - OS protocol and port attack defense

- Application security

  The following measures are used to ensure normal running of big data services:

  - Identification and authentication
  - Web application security
  - Access control
  - Audit security
  - Password security

- Data security

  The following measures are provided to ensure the confidentiality, integrity, and availability of massive amounts of user data:

  - Disaster recovery: MRS supports data backup to OBS and cross-region high reliability.
  - Backup: MRS supports backup of DBService, NameNode, and LDAP metadata and backup of HDFS and HBase service data.

- Data integrity

  Data is verified to ensure its integrity during storage and transmission.

  - CRC32C is used by default to verify the correctness of user data stored in HDFS.
  - DataNodes of HDFS store the verified data. If the data transmitted from a client is abnormal (incomplete), DataNodes report the abnormality to the client, and the client rewrites the data.
  - The client checks data integrity when reading data from a DataNode. If the data is incomplete, the client will read data from another DataNode.

- Data confidentiality

  Based on Apache Hadoop, the distributed file system of MRS supports encrypted storage of files to prevent sensitive data from being stored in plaintext, improving data security. Applications need only to encrypt specified sensitive data. Services are not affected during the encryption process. Based on file system data encryption, Hive provides table-level encryption and HBase provides column family-level encryption. Sensitive data can be encrypted and stored after you specify an encryption algorithm during table creation.

  Encrypted storage and access control of data are used to ensure user data security.

  - HBase stores service data to the HDFS after compression. Users can configure the AES and SMS4 encryption algorithm to encrypt data.
  - All the components allow access permissions to be set for local data directories. Unauthorized users are not allowed to access data.
  - All cluster user information is stored in ciphertext.

- Security authentication

  - Uses a unified user- and role-based authentication system as well as an account- and role-based access control (RBAC) model to centrally control user permissions and batch manage user authorization.

  - Employs Lightweight Directory Access Protocol (LDAP) as an account management system and performs the Kerberos authentication on accounts.
  - Provides the single sign-on (SSO) function that centrally manages and authenticates MRS system and component users.
  - Audits users who have logged in to Manager.

1.5.3 Easy Access to Web UIs of Components

Big data components have their own web UIs to manage their own systems. However, you cannot easily access the web UIs due to network isolation. For example, to access the HDFS web UI, you need to create an ECS to remotely log in to the web UI. This makes the UI access complex and unfriendly.

MRS provides an EIP-based secure channel for you to easily access the web UIs of components. This is more convenient than binding an EIP by yourself, and you can access the web UIs with a few clicks, avoiding the steps for logging in to a VPC, adding security group rules, and obtaining a public IP address. For the Hadoop, Spark, HBase, and Hue components in analysis clusters and the Storm component in streaming clusters, you can quickly access their web UIs from the entries on Manager.

1.5.4 Reliability Enhancement

Based on Apache Hadoop open source software, MRS optimizes and improves the reliability and performance of main service components.

System Reliability

- HA for all management nodes

  In the Hadoop open source version, data and compute nodes are managed in a distributed system, in which a single point of failure (SPOF) does not affect the operation of the entire system. However, a SPOF may occur on management nodes running in centralized mode, which becomes the weakness of the overall system reliability.

  MRS provides similar double-node mechanisms for all management nodes of the service components, such as Manager, HDFS NameNodes, HiveServers, HBase HMasters, Yarn ResourceManagers, KerberosServers, and LdapServers. All of them are deployed in active/standby mode or configured with load sharing, effectively preventing SPOFs from affecting system reliability.

- Reliability guarantee in case of exceptions

  By reliability analysis, the following measures to handle software and hardware exceptions are provided to improve the system reliability:

  - After power supply is restored, services are running properly regardless of a power failure of a single node or the whole cluster, ensuring data reliability in case of unexpected power failures. Key data will not be lost unless the hard disk is damaged.
  - Health status checks and fault handling of the hard disk do not affect services.
  - The file system faults can be automatically handled, and affected services can be automatically restored.

  - The process and node faults can be automatically handled, and affected services can be automatically restored.
  - The network faults can be automatically handled, and affected services can be automatically restored.
- Data backup and restoration

  MRS provides full backup, incremental backup, and restoration functions based on service requirements, preventing the impact of data loss and damages on services and ensuring fast system restoration in case of exceptions.

  - Automatic backup

    MRS provides automatic backup for data on Manager. Based on the customized backup policy, data on clusters, including LdapServer and DBService data, can be automatically backed up.
  - Manual backup

    You can also manually back up data of the cluster management system before the capacity expansion and patch installation to recover the cluster management system functions upon faults.

    To improve the system reliability, data on Manager and HBase is backed up to a third-party server manually.

Node Reliability

- OS health status monitoring

  MRS periodically collects OS hardware resource usage data, including usage of CPUs, memory, hard disks, and network resources.
- Process health status monitoring

  MRS checks the status of service instances and health indicators of service instance processes, enabling you to know the health status of processes in a timely manner.
- Automatic disk troubleshooting

  MRS is enhanced based on the open source version. It can monitor the status of hardware and file systems on all nodes. If an exception occurs, the corresponding partitions will be removed from the storage pool. If a disk is faulty and replaced, a new hard disk will be added for running services. In this case, maintenance operations are simplified. Replacement of faulty disks can be completed online. In addition, users can set hot backup disks to reduce the faulty disk restoration time and improve the system reliability.
- LVM configuration for node disks

  MRS allows you to configure Logic Volume Management (LVM) to plan multiple disks as a logical volume group. Configuring LVM can avoid uneven usage of disks. It is especially important to ensure even usage of disks on components that can use multiple disk capabilities, such as HDFS and Kafka. In addition, LVM supports disk capacity expansion without re-attaching, preventing service interruption.

Data Reliability

MRS can use the anti-affinity node groups and placement group capabilities provided by ECS and the rack awareness capability of Hadoop to redundantly

distribute data to multiple physical host machines, preventing data loss caused by physical hardware failures.

1.5.5 Job Management

The job management function provides an entry for you to submit jobs in a cluster, including MapReduce, Spark, HiveQL, and SparkSQL jobs. MRS works with Data Lake Factory (DLF) to provide a one-stop big data collaboration development environment and fully-managed big data scheduling capabilities, helping you effortlessly build big data processing centers.

DLF allows you to develop and debug MRS HiveQL/SparkSQL scripts online and develop MRS jobs by performing drag-and-drop operations to migrate and integrate data between MRS and more than 20 heterogeneous data sources. Powerful job scheduling and flexible monitoring and alarming help you easily manage data and job O&M.

1.5.6 Bootstrap Actions

Feature Introduction

MRS provides standard elastic big data clusters on the cloud. Nine big data components, such as Hadoop and Spark, can be installed and deployed. Currently, standard cloud big data clusters cannot meet all user requirements, for example, in the following scenarios:

- Common operating system configurations cannot meet data processing requirements, for example, increasing the maximum number of system connections.
- Software tools or running environments need to be installed, for example, Gradle and dependency R language package.
- Big data component packages need to be modified based on service requirements, for example, modifying the Hadoop or Spark installation package.
- Other big data components that are not supported by MRS need to be installed.

To meet the preceding customization requirements, you can manually perform operations on the existing and newly added nodes. The overall process is complex and error-prone. In addition, manual operations cannot be traced, and data cannot be processed immediately after creating a cluster based on your demand.

Therefore, MRS supports custom bootstrap actions that enable you to run scripts on a specified node before or after a cluster component is started. You can run bootstrap actions to install third-party software that is not supported by MRS, modify the cluster running environment, and perform other customizations. If you choose to run bootstrap actions when expanding a cluster, the bootstrap actions will be run on the newly added nodes in the same way. MRS runs the script you specify as user **root**. You can run the **su - *xxx*** command in the script to switch the user.

Customer Benefits

You can use the custom bootstrap actions to flexibly and easily configure your dedicated clusters and customize software installation.

1.5.7 Enterprise Project Management

An enterprise project is a cloud resource management mode. Enterprise Management provides users with comprehensive management of cloud-based resources, personnel, permissions, and finances. Common management consoles are oriented to the control and configuration of individual cloud products. The Enterprise Management console, in contrast, is more focused on resource management. It is designed to help enterprises manage cloud-based resources, personnel, permissions, and finances, in a hierarchical management manner, such as management of companies, departments, and projects.

MRS allows users who have enabled Enterprise Project Management Service (EPS) to configure enterprise projects for a cluster during cluster creation and use EPS to manage MRS resources by group.

- The users can manage multiple resources by group.
- The users can view resource information and expenditure details of enterprise projects.
- The users can control access permissions at the enterprise project level.
- The users can view detailed financial information by enterprise project, including orders, expenditure summary, and expenditure details.

1.5.8 Metadata

MRS provides multiple metadata storage methods. When deploying Hive and Ranger during MRS cluster creation, select one of the following storage modes as required:

- **Local**: Metadata is stored in the local GaussDB of a cluster. When the cluster is deleted, the metadata is also deleted. To retain the metadata, manually back up the metadata in the database in advance.
- **Data Connection**: Metadata is stored in the associated PostgreSQL or MySQL database of the RDS service in the same VPC and subnet as the current cluster. When the cluster is terminated, the metadata is not deleted. Multiple MRS clusters can share the metadata.

NOTE

Hive in MRS 1.9.*x* or later allows you to specify a metadata storage method.

Ranger in MRS 1.9.*x* allows metadata to be stored only in the associated MySQL database of the RDS service.

1.5.9 Cluster Management

1.5.9.1 Cluster Lifecycle Management

MRS supports cluster lifecycle management, including creating and terminating clusters.

- Creating a cluster: After you specify a cluster type, components, number of nodes of each type, VM specifications, AZ, VPC, and authentication information, MRS automatically creates a cluster that meets the configuration requirements. You can run customized scripts in the cluster. In addition, you can create clusters of different types for multiple application scenarios, such as Hadoop analysis clusters, HBase clusters, and Kafka clusters. The big data platform supports heterogeneous cluster deployment. That is, VMs of different specifications can be combined in a cluster based on CPU types, disk capacities, disk types, and memory sizes. Various VM specifications can be mixed in a cluster.
- Terminating a cluster: You can terminate a cluster that is no longer needed (including data and configurations in the cluster). MRS will delete all resources related to the cluster.

Creating a Cluster

On the MRS management console, you can create an MRS cluster. You can select a region and cloud resource specifications to create an MRS cluster that is suitable for enterprise services in one click. MRS automatically installs and deploys the enterprise-level big data platform and optimizes parameters based on the selected cluster type, version, and node specifications.

MRS provides you with fully managed big data clusters. When creating a cluster, you can set a VM login mode (password or key pair). You can use all resources of the created MRS cluster. In addition, MRS allows you to deploy a big data cluster on only two ECSs with 4 vCPUs and 8 GB memory, providing more flexible choices for testing and development.

MRS clusters are classified into analysis, streaming, and hybrid clusters.

- Analysis cluster: is used for offline data analysis and provides Hadoop components.
- Streaming cluster: is used for streaming tasks and provides stream processing components.
- Hybrid cluster: is used for not only offline data analysis but also streaming processing, and provides Hadoop components and stream processing components.
- Custom: You can flexibly combine required components (MRS 3.x and later versions) based on service requirements.

MRS cluster nodes are classified into Master, Core, and Task nodes.

- Master node: management node in a cluster. Master processes of a distributed system, Manager, and databases are deployed on Master nodes. Master nodes cannot be scaled out. The processing capability of Master nodes determines the upper limit of the management capability of the entire cluster. MRS supports scale-up of Master node specifications to provide support for management of a larger cluster.
- Core node: used for both storage and computing and can be scaled in or out. Since Core nodes bear data storage, there are many restrictions on scale-in to prevent data loss and auto scaling cannot be performed.
- Task node: used only for computing only and can be scaled in or out. Task nodes bear only computing tasks. Therefore, auto scaling can be performed.

You can create a cluster in two modes: custom create a cluster and quick create a cluster.

- **Custom config**: On the **Custom Config** page, you can flexibly configure cluster parameters based on application scenarios, such as ECS specifications to better suit your service requirements.
- **Quick config**: On the **Quick Config** page, you can quickly create a cluster based on application scenarios, improving cluster configuration efficiency. Currently, Hadoop analysis clusters, HBase clusters, and Kafka clusters are available for your quick creation.
  - Hadoop analysis cluster: uses components in the open-source Hadoop ecosystem to analyze and query vast amounts of data. For example, use Yarn to manage cluster resources, Hive and Spark to provide offline storage and computing of large-scale distributed data, Spark Streaming and Flink to offer streaming data computing, and Presto to enable interactive queries, and Tez to provide a distributed computing framework of directed acyclic graphs (DAGs).
  - HBase cluster: uses Hadoop and HBase components to provide a column-oriented distributed cloud storage system featuring enhanced reliability, excellent performance, and elastic scalability. It applies to the storage and distributed computing of massive amounts of data. You can use HBase to build a storage system capable of storing TB- or even PB-level data. With HBase, you can filter and analyze data with ease and get responses in milliseconds, rapidly mining data value.
  - Kafka cluster: uses Kafka and Storm to provide an open source message system with high throughput and scalability. It is widely used in scenarios such as log collection and monitoring data aggregation to implement efficient streaming data collection and real-time data processing and storage.

Terminating a Cluster

MRS allows you to terminate a cluster when it is no longer needed. After the cluster is terminated, all cloud resources used by the cluster will be released. Before terminating a cluster, you are advised to migrate or back up data. Terminate the cluster only when no service is running in the cluster or the cluster is abnormal and cannot provide services based on O&M analysis. If data is stored on EVS disks or pass-through disks in a big data cluster, the data will be deleted after the cluster is terminated. Therefore, exercise caution when terminating a cluster.

1.5.9.2 Manually Scale Out/In a Cluster

The processing capability of a big data cluster can be horizontally expanded by adding nodes. If the cluster scale does not meet service requirements, you can manually scale out or scale in the cluster. MRS intelligently selects the node with the least load or the minimum amount of data to be migrated for scale-in. The node to be scaled in will not receive new tasks, and continues to execute the existing tasks. At the same time, MRS copies its data to other nodes and the node is decommissioned. If the tasks on the node cannot be completed after a long time, MRS migrates the tasks to other nodes, minimizing the impact on cluster services.

Scaling Out a Cluster

Currently, you can add Core or Task nodes to scale out a cluster to handle peak service loads. The capacity expansion of an MRS cluster node does not affect the services of the existing cluster.

Scaling In a Cluster

You can reduce the number of Core or Task nodes to scale in a cluster so that MRS delivers better storage and computing capabilities at lower O&M costs based on service requirements. After you scale in an MRS cluster, MRS automatically selects nodes that can be scaled in based on the type of services installed on the nodes.

During the scale-in of Core nodes, data on the original nodes is migrated. If the data location is cached, the client automatically updates the location information, which may affect the latency. Node scale-in may affect the response duration of the first access to some HBase on HDFS data. You can restart HBase or disable or enable related tables to avoid this problem.

Task nodes do not store cluster data. They are compute nodes and do not involve migration of data on the nodes.

1.5.9.3 Auto Scaling

Feature Introduction

More and more enterprises use technologies such as Spark and Hive to analyze data. Processing a large amount of data consumes huge resources and costs much. Typically, enterprises regularly analyze data in a fixed period of time every day rather than all day long. To meet enterprises' requirements, MRS provides the auto scaling function to apply for extra resources during peak hours and release resources during off-peak hours. This enables users to use resources on demand and focus on core business at lower costs.

In big data applications, especially in periodic data analysis and processing scenarios, cluster computing resources need to be dynamically adjusted based on service data changes to meet service requirements. The auto scaling function of MRS enables clusters to be elastically scaled out or in based on cluster loads. In addition, if the data volume changes regularly and you want to scale out or in a cluster before the data volume changes, you can use the MRS resource plan feature.

MRS supports two types of auto scaling policies: auto scaling rules and resource plans

- Auto scaling rules: You can increase or decrease Task nodes based on real-time cluster loads. Auto scaling will be triggered when the data volume changes but there may be some delay.
- Resource plans: If the data volume changes periodically, you can create resource plans to resize the cluster before the data volume changes, thereby avoiding a delay in increasing or decreasing resources.

Both auto scaling rules and resource plans can trigger auto scaling. You can configure both of them or configure one of them. Configuring both resource plans

and auto scaling rules improves the cluster node scalability to cope with occasionally unexpected data volume peaks.

In some service scenarios, resources need to be reallocated or service logic needs to be modified after cluster scale-out or scale-in. If you manually scale out or scale in a cluster, you can log in to cluster nodes to reallocate resources or modify service logic. If you use auto scaling, MRS enables you to customize automation scripts for resource reallocation and service logic modification. Automation scripts can be executed before and after auto scaling and automatically adapt to service load changes, all of which eliminates manual operations. In addition, automation scripts can be fully customized and executed at various moments, which can meet your personalized requirements and improve auto scaling flexibility.

Customer Benefits

MRS auto scaling provides the following benefits:

- Reducing costs

  Enterprises do not analyze data all the time but perform a batch data analysis in a specified period of time, for example, 03:00 a.m. The batch analysis may take only two hours.

  The auto scaling function enables enterprises to add nodes for batch analysis and automatically releases the nodes after completion of the analysis, minimizing costs.

- Meeting instant query requirements

  Enterprises usually encounter instant analysis tasks, for example, data reports for supporting enterprise decision-making. As a result, resource consumption increases sharply in a short period of time. With the auto scaling function, computing nodes can be added for emergent big data analysis, avoiding a service breakdown due to insufficient computing resources. You do not need to purchase extra resources. After the emergency event ends, MRS can automatically release the nodes.

- Focusing on core business

  It is difficult for developers to determine resource consumption on the big data secondary development platform because of complex query analysis conditions (such as global sorting, filtering, and merging) and data complexity, for example, uncertainty of incremental data. As a result, estimating the computing volume is difficult. MRS's auto scaling function enable developers to focus on service development without the need for resource estimation.

1.5.9.4 Task Node Creation

Feature Introduction

Task nodes can be created and used for computing only. They do not store persistent data and are the basis for implementing auto scaling.

Customer Benefits

When MRS is used only as a computing resource, Task nodes can be used to reduce costs and facilitate cluster node scaling, flexibly meeting users' requirements for increasing or decreasing cluster computing capabilities.

Application Scenarios

When the data volume change is small in a cluster but the cluster's service processing capabilities need to be remarkably and temporarily improved, add Task nodes to address the following situations:

- The number of temporary services is increased, for example, report processing at the end of the year.
- Long-term tasks need to be completed in a short time, for example, some urgent analysis tasks.

1.5.9.5 Scaling Up Master Node Specifications

MRS provides Manager for managing clusters and services in the clusters, such as NameNodes of HDFS, ResourceManagers of Yarn, and Manager management services of MRS, are deployed on the Master node of the clusters.

With the rollout of new services, a cluster scale increases continuously, and Master nodes bear more and more loads. Enterprise users are faced with the problem that CPU loads are too high and memory usage exceeds the threshold. Generally, in an on-premises big data cluster, you need to migrate data and purchase hardware with advanced configurations to scale up the Master node specifications. MRS leverages the advantages of cloud services to enable you to scale up Master node specifications in one click. During the scale-up, the active/standby HA mode of the Master nodes ensures that existing services are not interrupted.

For details about how to scale up the Master node specifications, see **Scaling Up Master Node Specifications**.

1.5.9.6 Isolating a Host

When detecting that a host is abnormal or faulty and cannot provide services or affects cluster performance, you can exclude the host from the available nodes in the cluster temporarily so that the client can access other available nodes. In scenarios where patches are to be installed in a cluster, you can also exclude a specified node from patch installation. Only non-management nodes can be isolated.

After a host is isolated, all role instances on the host will be stopped, and you cannot start, stop, or configure the host and all instances on the host. In addition, after a host is isolated, statistics about the monitoring status and metric data of hardware and instances on the host cannot be collected or displayed.

1.5.9.7 Managing Tags

Tags are cluster identifiers. Adding tags to clusters can help you identify and manage your cluster resources. By associating with Tag Management Service (TMS), MRS allows users with a large number of cloud resources to tag cloud resources, quickly search for cloud resources with the same tag attribute, and

perform unified management operations such as review, modification, and deletion, facilitating unified management of big data clusters and other cloud resources.

You can add a maximum of 10 tags to a cluster when creating the cluster or add them on the details page of the created cluster.

1.5.10 Cluster O&M

Alarm Management

MRS can monitor big data clusters in real time and identify system health status based on alarms and events. In addition, MRS allows you to customize monitoring and alarm thresholds to focus on the health status of each metric. When monitoring data reaches the alarm threshold, the system triggers an alarm.

MRS can also interconnect with the message service system of the Simple Message Notification (SMN) service to push alarm information to users by SMS message or email. For details, see **Message Notification**.

Patch Management

MRS supports cluster patching operations and will release patches for open source big data components in a timely manner. On the MRS cluster management page, you can view patch release information related to running clusters, including the detailed description of the resolved issues and impacts. You can determine whether to install a patch based on the service running status. One-click patch installation involves no manual intervention, and will not cause service interruption through rolling installation, ensuring long-term availability of the clusters.

MRS can display the detailed patch installation process. Patch management also supports patch uninstallation and rollback.

NOTE

MRS 3.x or later does not support patch management on the management console.

O&M Support

Cluster resources provided by MRS belong to users. Generally, when O&M personnel's support is required for troubleshooting of a cluster, O&M personnel cannot directly access the cluster. To better serve customers, MRS provides the following two methods to improve communication efficiency during fault locating:

- Log sharing: You can initiate log sharing on the MRS management console to share a specified log scope with O&M personnel, so that O&M personnel can locate faults without accessing the cluster.
- O&M authorization: If a problem occurs when you use an MRS cluster, you can initiate O&M authorization on the MRS management console. O&M personnel can help you quickly locate the problem, and you can revoke the authorization at any time.

Health Check

MRS provides automatic inspection on system running environments for you to check and audit system running health status in one click, ensuring proper system running and lowering system operation and maintenance costs. After viewing inspection results, you can export reports for archiving and fault analysis.

1.5.11 Message Notification

Feature Introduction

The following operations are often performed during the running of a big data cluster:

- Big data clusters often change, for example, cluster scale-out and scale-in.
- When a service data volume changes abruptly, auto scaling will be triggered.
- After related services are stopped, a big data cluster needs to be stopped.

To immediately notify you of successful operations, cluster unavailability, and node faults, MRS uses Simple Message Notification (SMN) to send notifications to you through SMS and emails, facilitating maintenance.

Customer Benefits

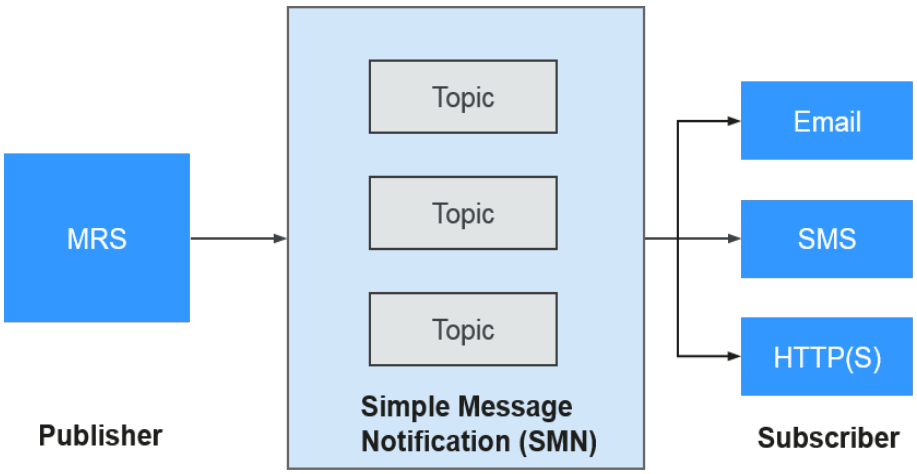
After configuring SMN, you can receive MRS cluster health status, updates, and component alarms through SMS or emails in real time. MRS sends real-time monitoring and alarm notification to help you easily perform O&M and efficiently deploy big data services.

Feature Description

MRS uses SMN to provide one-to-multiple message subscription and notification over a variety of protocols.

You can create a topic and configure topic policies to control publisher and subscriber permissions on the topic. MRS sends cluster messages to the topic to which you have permission to publish messages. Then, all subscribers who subscribe to the topic can receive cluster updates and component alarms through SMS and emails.

Figure 1-146 Implementation process



1.6 Constraints

Before using MRS, ensure that you have read and understand the following restrictions.

- MRS clusters must be created in VPC subnets.
- You are advised to use any of the following browsers to access MRS:
  - Google Chrome: 36.0 or later
  - Internet Explorer: 9.0 or later

    If you use Internet Explorer 9.0, you may fail to log in to the MRS management console because user **Administrator** is disabled by default in some Windows systems, such as Windows 7 Ultimate. Internet Explorer automatically selects a system user for installation. As a result, Internet Explorer cannot access the management console. Reinstall Internet Explorer 9.0 or later (recommended) or run Internet Explorer 9.0 as user **Administrator**.
- When you create an MRS cluster, you can select **Auto create** from the drop-down list of **Security Group** to create a security group or select an existing security group. After the MRS cluster is created, do not delete or modify the used security group. Otherwise, a cluster exception may occur.
- To prevent illegal access, only assign access permission for security groups used by MRS where necessary.
- Do not perform the following operations because they will cause cluster exceptions:
  - Shutting down, restarting, or deleting MRS cluster nodes displayed in ECS, changing or reinstalling their OS, or modifying their specifications.
  - Deleting the existing processes, applications or files on cluster nodes.
- If a cluster exception occurs when no incorrect operations have been performed, contact technical support engineers. They will ask you for your password and then perform troubleshooting.
- Plan disks of cluster nodes based on service requirements. If you want to store a large volume of service data, add EVS disks or storage space to prevent insufficient storage space from affecting node running.
- The cluster nodes store only users' service data. Non-service data can be stored in the OBS or other ECS nodes.
- The cluster nodes only run MRS cluster programs. Other client applications or user service programs are deployed on separate ECS nodes.
- The storage capacity of MRS cluster nodes (including master, core, and task nodes) can be expanded only by attaching new disks instead of expanding capacity of the existing disks.
- If the cluster is still used to execute tasks or modify configurations after a master node in the cluster has been stopped, and other master nodes in the cluster are stopped before the stopped master node is started after the task execution or configuration modification, data may be lost due to an active/standby switchover. In this scenario, after the task is executed or the configuration is modified, start the master node that has been stopped and

then stop all nodes. If all nodes in the cluster have been stopped, start them in the reverse order of node shutdown.

- The Capacity and Superior scheduler switchover is complete when the MRS cluster is used, while configuration synchronization is not complete. Configure synchronization again based on the new scheduler if necessary.

1.7 Technical Support

MRS is a one-stop big data platform that provides enterprise-class clusters on the cloud. Tenants can fully control clusters and easily run big data components such as Hadoop, Hive, Spark, HBase, Kafka, and Flink. In addition, MRS helps enterprises quickly build a system to process massive amounts of data and discover new value and business opportunities in real time or in non-real time.

Maintenance Policy Statement

MRS provides tenants with fully controllable clusters and semi-hosting cloud services. By default, cloud services do not have permissions to perform operations on the clusters. Tenants are responsible for routine cluster O&M and management. They can contact technical support team for help if any technical issues occur, excluding those not related to MRS, for example, how to build an application system based on the big data platform.

Technical Support Scope

- The MRS console provides the following functions:
  - Creating, deleting, and scaling in or out a cluster
  - Managing cluster jobs
  - Managing cluster alarms
  - Managing cluster patches
  - Managing IAM users
  - Managing external APIs
- MRS supports vulnerability analysis of open-source components, such as impact analysis and fixing suggestions. It enables tenants to evaluate the impact of vulnerabilities on services and fix the vulnerabilities.

1.8 Permissions Management

If you need to assign different permissions to employees in your enterprise to access your MRS resources in the cloud, IAM is a good choice for fine-grained permissions management. IAM provides identity authentication, permissions management, and access control, helping you secure access to your cloud resources.

With IAM, you can create IAM users under your cloud account, and assign permissions to these users to control their access to specific resources. For example, some software developers in your enterprise need to use MRS resources but must not delete MRS clusters or perform any high-risk operations. To achieve this goal, you can create IAM users for the software developers and grant them only the permissions required for using MRS cluster resources.

If your cloud account does not require individual IAM users for permissions management, skip this section.

IAM is free of charge.

MRS Permission Description

By default, new IAM users do not have any permissions. To assign permissions to a user, add the user to one or more groups and assign permissions policies or roles to these groups. The user then inherits permissions from the groups it is a member of and can perform specified operations on cloud services based on the permissions.

MRS is a project-level service deployed and accessed in specific physical regions. To assign permissions to a user group, specify **Scope** as **Region-specific projects** and select projects in the corresponding region for the permissions to take effect. If **All projects** is selected, the permissions will take effect for the user group in all region-specific projects. When accessing MRS, the users need to switch to a region where they have been authorized to use the MRS service.

You can grant users permissions by using roles and policies.

- Roles: A type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. This mechanism provides only a limited number of service-level roles for authorization. When using roles to grant permissions, you need to also assign other roles on which the permissions depend to take effect. However, roles are not an ideal choice for fine-grained authorization and secure access control.
- Policies: A type of fine-grained authorization mechanism that defines permissions required to perform operations on specific cloud resources under certain conditions. This mechanism allows for more flexible policy-based authorization, meeting requirements for secure access control. For example, you can grant MRS users only the permissions for performing specified operations on MRS clusters, such as creating a cluster and querying a cluster list rather than deleting a cluster. Most policies define permissions based on APIs.

Table 1-29 lists all the system policies supported by MRS.

Table 1-29 MRS system policies

Policy	Description	Type
MRS FullAccess	Administrator permissions for MRS. Users granted these permissions can operate and use all MRS resources.	Fine-grained policy
MRS CommonOperations	Common user permissions for MRS. Users granted these permissions can use MRS but cannot add or delete resources.	Fine-grained policy
MRS ReadOnlyAccess	Read-only permission for MRS. Users granted these permissions can only view MRS resources.	Fine-grained policy

Policy	Description	Type
MRS Administrator	Permissions: • All operations on MRS • Users with permissions of this policy must also be granted permissions of the **Tenant Guest**, **Server Administrator**, and **BSS Administrator** policies.	RBAC policy

Table 1-30 lists the common operations supported by each system-defined policy or role of MRS. Select the policies or roles as required.

Table 1-30 Common operations supported by each system-defined policy

Operation	MRS FullAccess	MRS CommonOperations	MRS ReadOnlyAccess	MRS Administrator
Creating a cluster	√	x	x	√
Resizing a cluster	√	x	x	√
Upgrading node specifications	√	x	x	√
Deleting a cluster	√	x	x	√
Querying cluster details	√	√	√	√
Querying a cluster list	√	√	√	√
Configuring an auto scaling rule	√	x	x	√
Querying a host list	√	√	√	√
Querying operation logs	√	√	√	√

Operation	MRS FullAccess	MRS CommonOperations	MRS ReadOnlyAccess	MRS Administrator
Creating and executing a job	√	√	x	√
Stopping a job	√	√	x	√
Deleting a single job	√	√	x	√
Deleting jobs in batches	√	√	x	√
Querying job details	√	√	√	√
Querying a job list	√	√	√	√
Creating a folder	√	√	x	√
Deleting a file	√	√	x	√
Querying a file list	√	√	√	√
Operating cluster tags in batches	√	√	x	√
Creating a single cluster tag	√	√	x	√
Deleting a single cluster tag	√	√	x	√
Querying a resource list by tag	√	√	√	√
Querying cluster tags	√	√	√	√
Accessing Manager	√	√	x	√

Operation	MRS FullAccess	MRS CommonOperations	MRS ReadOnlyAccess	MRS Administrator
Querying a patch list	√	√	√	√
Installing a patch	√	√	x	√
Uninstalling a patch	√	√	x	√
Authorizing O&M channels	√	√	x	√
Sharing O&M channel logs	√	√	x	√
Querying an alarm list	√	√	√	√
Subscribing to alarm notification	√	√	x	√
Submitting an SQL statement	√	√	x	√
Querying SQL results	√	√	x	√
Canceling an SQL execution task	√	√	x	√

1.9 Related Services

Relationships with Other Services

Table 1-31 Relationships with other services

Service	Relationships
Virtual Private Cloud (VPC)	MRS clusters are created in the subnets of a VPC. VPCs provide a secure, isolated, and logical network environment for your MRS clusters.
Object Storage Service (OBS)	OBS stores the following user data: • MRS job input data, such as user programs and data files • MRS job output data, such as result files and log files of jobs In MRS clusters, HDFS, Hive, MapReduce, YARN, Spark, Flume, and Loader can import or export data from OBS. MRS uses the parallel file system of OBS to provide services.
Elastic Cloud Server (ECS)	MRS uses elastic cloud servers (ECSs) as cluster nodes.
Relational Database Service (RDS)	RDS stores MRS system running data, including MRS cluster metadata.
Identity and Access Management (IAM)	IAM provides authentication for MRS.
Simple Message Notification (SMN)	MRS uses SMN to provide one-to-multiple message subscription and notification over a variety of protocols.
Cloud Trace Service (CTS)	CTS provides you with operation records of MRS resource operation requests and request results for querying, auditing, and backtracking.

Table 1-32 MRS operations recorded by CTS

Operation	Resource Type	Trace Name
Creating a cluster	cluster_mrs	createCluster
Deleting a cluster	cluster_mrs	deleteCluster
Expanding a cluster	cluster_mrs	scaleOutCluster
Shrinking a cluster	cluster_mrs	scaleInCluster

After you enable CTS, the system starts recording operations on cloud resources. You can view operation records of the last 7 days on the CTS management

console. For details, see **Cloud Trace Service** > **Getting Started** > **Querying Real-Time Traces**.

1.10 Common Concepts

HBase Table

An HBase table is a three-dimensional map comprised of one or more columns or rows of data.

Column

Column is a dimension of an HBase table. The column name is in the format of *<family>*:*<label>*, where *<family>* and *<label>* can be any combination of characters. An HBase table consists of a set of column families. Each column in the HBase table belongs to a column family.

Column Family

A column family is a collection of columns stored in the HBase schema. To create columns, you must create a column family first. A column family organizes data with the same property in HBase. Each row of data in the same column family is stored on the same server. Each column family can be one attribute, such as compressed packages, timestamps, and data block cache.

MemStore

MemStore is a core of HBase storage. When the amount of data stored in WAL reaches the upper limit, the data is loaded to MemStore for sorting and storage.

RegionServer

RegionServer is a service running on each DataNode in the HBase cluster. It is responsible for serving and managing regions, uploading the load information of regions, and managing distributed master nodes.

Timestamp

A timestamp is a 64-bit integer used to index different versions of the same data. A timestamp can be automatically assigned by HBase when data is written or assigned by users.

Store

Store is a core of HBase storage. A Store hosts one MemStore and multiple StoreFiles. A Store corresponds to a column family of a table in a region.

Index

An index is a data structure that improves the efficiency of data retrieval in a database table. One or more columns in a database table can be used for fast random retrieval of data and efficient access to ordered records.

Coprocessor

A coprocessor is an interface provided by HBase for implementing calculation logic on RegionServer. Coprocessors are classified into system coprocessors and table coprocessors. The former can import all data tables on RegionServer, and the latter can process a specified table.

Block Pool

A block pool is a collection of blocks that belong to a single namespace. DataNodes store blocks from all block pools in a cluster. Each block pool is managed independently, which allows a namespace to generate an ID for a new block without relying on other namespaces. If one NameNode is invalid, the DataNode can still provide services for other NameNodes in the cluster.

DataNode

A DataNode is a worker node in the HDFS cluster. Scheduled by the client or NameNode, DataNodes store and retrieve data and periodically report file blocks to NameNodes.

File Block

A file block is the minimum logical unit stored in the HDFS. Each HDFS file is stored in one or more file blocks. All file blocks are stored in DataNodes.

Block Replica

A replica is a block copy stored in HDFS. A file block stores multiple replicas for system availability and fault tolerance.

Namespace Volume

A namespace volume is an independent management unit that consists of a namespace and its block pool. When a NameNode or namespace is deleted, the related block pools on the DataNode are also deleted. During a cluster upgrade, each namespace volume is upgraded as a whole.

NodeManager

NodeManager executes applications, monitors the usage of resources (including CPUs, memory, disks, and network resources) of applications, and reports the resource usage to the ResourceManager.

ResourceManager

ResourceManager schedules resources required by applications. It provides a scheduling plug-in for allocating cluster resources to multiple queues and applications. The scheduling plug-in schedules resources based on existing capabilities or using the fair scheduling model.

Partition

Each topic can be divided into multiple partitions. Each partition corresponds to an appendant log file whose sequence is fixed.

Follower

A follower processes read requests and works with a leader to process write requests. It can also be used as a leader backup. When the leader is faulty, a follower is elected to take over the leader's workload to prevent a single point of failure.

Observer

Observers do not take part in voting for election and write requests. They only process read requests and forward write requests to the leader, improving processing efficiency.

Leader

A leader of the ZooKeeper clusters is elected by followers using the Zookeeper Atomic Broadcast (ZAB) protocol. It receives and schedules all write requests, and synchronizes written information to followers and observers.

CarbonData

A Carbon is an open architecture based on Spark SQL. It integrates the developed MOLAP engine and Spark, and quickly builds the Spark-based distributed multi-dimensional analysis engine, shortening the analysis duration from minutes to seconds and strengthening the multi-dimensional analysis capability of Spark.

DStream

DStream is an abstract concept provided by Spark Streaming. It is a continuous data stream which is obtained from the data source or the transformed input stream. In essence, a DStream is a series of continuous resilient distributed datasets (RDDs).

Heap Memory

A heap indicates the data area where the Java Virtual Machine (JVM) is running and from which memory for all class instances and arrays is committed. The JVM startup parameters **-Xms** and **-Xmx** are used to set the initial heap memory and the maximum heap memory, respectively.

- Maximum heap memory: Heap memory that can be committed to a program at most by the system, which is specified by the **-Xmx** parameter.
- Committed heap memory: total heap memory committed by the system for running a program. It ranges from the initial heap memory and the maximum heap memory.
- Used heap memory: heap memory that has been used by a program. It is smaller than the committed heap memory.

- Non-heap memory: memory excluded from the JVM heaps and the memory area for running the JVM. Non-heap memory has the following three memory pools:
  - Code Cache: stores JIT compiled code. Its value is set through the JVM startup parameter **-XX:InitialCodeCacheSize -XX:ReservedCodeCacheSize**. The default value is 240 MB.
  - Compressed Class Space: stores metadata of a pointer. Its value is set through the JVM startup parameter **-XX:CompressedClassSpaceSize**. The default value is 1024 MB.
  - Metaspace: stores metadata. Its value is set through the JVM startup parameter **-XX:MetaspaceSize -XX:MaxMetaspaceSize**.
- Maximum non heap memory: non-heap memory committed to a program at most by the system. Its value is the sum of the maximum values of Code Cache, Compressed Class Space, and Metaspace.
- Committed non-heap memory: total non-heap memory committed by the system for running a program. It ranges from the initial non-heap memory and the maximum non-heap memory.
- Used non-heap memory: non-heap memory that has been used by aprogram. It is smaller than the committed non-heap memory.

Hadoop

Hadoop is a distributed system framework. It allows users to develop distributed applications using high-speed computing and storage provided by clusters without knowing the underlying details of the distributed system. It can also reliably and efficiently process massive amounts of data in scalable, distributed mode. Hadoop is reliable because it maintains multiple work data duplicates, enabling distributed processing for failed nodes. Hadoop is highly efficient because it processes data in parallel mode. Hadoop is scalable because it can process petabytes of data. Hadoop consists of HDFS, MapReduce, HBase, and Hive.

Role

A role is an element of a service. A service contains one or multiple roles. Services are installed on servers through roles so that they can run properly.

Cluster

A cluster is computer technology that enables multiple servers to work as one server. Clusters improve the stability, reliability, and data processing or service capability of the system. For example, clusters can prevent single point of failures (SPOFs), share storage resources, reduce system load, and improve system performance.

Instance

An instance is formed when a service role is installed on the host. A service has one or more role instances.

Metadata

Metadata is data that provides information about other data and is also called media data or relay data. It is used to define data properties, specify data storage locations and historical data, retrieve resources, and record files.

2 MRS Quick Start

2.1 How to Use MRS

MapReduce Service (MRS) is a cloud service that is used to deploy and manage Hadoop systems and enables one-click Hadoop cluster deployment. MRS provides Hadoop-based high-performance big data components, such as Hadoop, Spark, HBase, Kafka, and Storm.

MRS is easy to use. You can execute various tasks and process or store PB-level data using computers connected in a cluster. The procedure of using MRS is as follows:

1. Upload local programs and data files to OBS.
2. Create a cluster by following instructions in **Creating a Custom Cluster**. You can choose a cluster type for offline data analysis or stream processing or both, and set ECS instance specifications, instance count, data disk type (common I/O, high I/O, and ultra-high I/O), and components to be installed such as Hadoop, Spark, HBase, Hive, Kafka, and Storm in a cluster. You can use a **bootstrap action** to execute a script on a specified node before or after the cluster is started to install additional third-party software, modify the cluster running environment, and perform other customizations.
3. **Manage jobs**. MRS provides a platform for executing programs you develop. You can submit, execute, and monitor such programs on MRS.
4. **Manage clusters**. MRS provides you with MRS Manager, an enterprise-level unified management platform of big data clusters, helping you quickly know health status of services and hosts. Through graphical metric monitoring and customization, you can obtain critical system information in a timely manner. In addition, you can modify service attribute configurations based on service performance requirements, and start or stop clusters, services, and role instances in one click.
5. **Terminating a Cluster**. You can terminate an MRS cluster that is no longer use after job execution is complete.

2.2 Creating a Cluster

The first step of using MRS is to a cluster. This section describes how to create a cluster on the MRS management console.

Procedure

Step 1 Log in to the MRS console.

Step 2

NOTE

When creating a cluster, pay attention to quota notification. If a resource quota is insufficient, increase the resource quota as prompted and create a cluster.

Step 3 On the page for a cluster, click the **Custom Config** tab.

Step 4 Configure cluster software information.

- **Region**: Use the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, for example, **mrs_20180321**.
- **Cluster Version**: Select the latest version, which is the default value.
- **Cluster Type**: Use the default **Analysis Cluster**.
- **Component**: Select components such as Spark2x, HBase, and Hive for the analysis cluster. For a streaming cluster, select components such as Kafka and Storm. For a hybrid cluster, you can select the components of the analysis cluster and streaming cluster based on service requirements.
- **Metadata**: Retain the default value.

  NOTE

  For versions earlier than MRS 3.x, select components such as Spark, HBase, and Hive for the analysis cluster.

Step 5 Click **Next**.

- **AZ**: Use the default value.
- **VPC**: Use the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Use the default value.
- **Security Group**: Select **Auto create**.
- **EIP**: Select **Bind later**.
- **Enterprise Project**: Use the default value.
- **Instance Specifications**: Select General Computing S3 -> 8 vCPUs | 16 GB (s3.2xlarge.2) for both Master and Core nodes.
- **System Disk**: Select **Common I/O** and retain the default settings.
- **Data Disk**: Select **Common I/O** and retain the default settings.
- **Instance Count**: The default number of Master nodes is 2, and that of Core nodes is 3.

Step 6 Click **Next**. The **Set Advanced Options** tab page is displayed. Configure the following parameters. Retain the default settings for the other parameters.

- Kerberos authentication:
  - **Kerberos Authentication**: Disable Kerberos authentication.
  - **Username**: name of the Manager administrator. **admin** is used by default.
  - **Password**: password of the Manager administrator.
- **Login Mode**: Select a mode for logging in to an ECS.
  - **Password**: Set a password for logging in to an ECS.
  - **Key Pair**: Select a key pair from the drop-down list. Select **"I acknowledge that I have obtained private key file *SSHkey-xxx* and that without this file I will not be able to log in to my ECS.**" If you have never created a key pair, click **View Key Pair** to create or import a key pair. And then, obtain a private key file.
- **Secure Communications**: Select **Enable**.

Step 7 Click **Apply Now**.

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 8 Click **Back to Cluster List** to view the cluster status.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

----End

2.3 Uploading Data and Programs

Through the **Files** tab page, you can create, delete, import, export, delete files in the analysis cluster.

Background

MRS clusters process data from OBS or HDFS. OBS provides customers with the data storage capabilities that are massive, secure, reliable, and cost-effective. MRS can directly process data in OBS. You can browse, manage, and use data on the web page of the management console and OBS Client.

Importing Data

Currently, MRS can only import data from OBS to HDFS. The file upload rate decreases with the increase of the file size. This mode applies to scenarios where the data volume is small.

You can perform the following steps to import files and directories:

1. Log in to the MRS console.

2. Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's information.
3. Click the **Files** tab to go to the file management page.
4. Select **HDFS File List**.
5. Go to the data storage directory, for example, **bd_app1**.

   The **bd_app1** directory is only an example. You can use any directory on the page or create a new one.

   The requirements for creating a folder are as follows:

   - The folder name contains a maximum of 255 characters
   - The folder name cannot be empty.
   - The folder name cannot contain the following special characters: /:*?"<>|\;&,'`!{}[]$%+
   - The value cannot start or end with a period (.).
   - The spaces at the beginning and end are ignored.
6. Click **Import Data** and configure the HDFS and OBS paths correctly. When configuring the OBS or HDFS path, click **Browse**, select a file directory, and click **Yes**.
   - OBS path
     - The path must start with **obs://**.
     - Files or programs encrypted by KMS cannot be imported.
     - An empty folder cannot be imported.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters ;|&>,<'$*?\
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The OBS full path contains a maximum of 255 characters.
   - HDFS path
     - The path starts with **/user** by default.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters: ;|&>,<'$*?\:
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The HDFS full path contains a maximum of 255 characters.
7. Click **OK**.

   You can view the file upload progress on the **File Operation Records** tab page. MRS processes the data import operation as a DistCp job. You can also check whether the DistCp job is successfully executed on the **Jobs** tab page.

Exporting Data

After the data analysis and computing are completed, you can store the data in HDFS or export them to OBS.

You can perform the following steps to export files and directories:

1. Log in to the MRS console.
2. Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.
3. Click the **Files** tab to go to the file management page.
4. Select **HDFS File List**.
5. Go to the data storage directory, for example, **bd_app1**.
6. Click **Export Data** and configure the OBS and HDFS paths. When configuring the OBS or HDFS path, click **Browse**, select a file directory, and click **Yes**.
   - OBS path
     - The path must start with **obs://**.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters ;|&>,<'$*?\
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The OBS full path contains a maximum of 255 characters.
   - HDFS path
     - The path starts with **/user** by default.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters: ;|&>,<'$*?\:
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The HDFS full path contains a maximum of 255 characters.

   NOTE

   When a folder is exported to OBS, a label file named **folder name_$folder$** is added to the OBS path. Ensure that the exported folder is not empty. If the exported folder is empty, OBS cannot display the folder and only generates a file named **folder name_$folder$**.
7. Click **OK**.

   You can view the file upload progress on the **File Operation Records** tab page. MRS processes the data export operation as a DistCp job. You can also check whether the DistCp job is successfully executed on the **Jobs** tab page.

2.4 Creating a Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results.

This section describes how to submit a job (take a MapReduce job as an example) on the MRS management console. MapReduce jobs are used to submit JAR programs to quickly process massive amounts of data in parallel and create a distributed data processing and execution environment.

If the job and file management functions are not supported on the cluster details page, submit the jobs in the background.

Before creating a job, you need to upload local data to OBS for data computing and analyzing. MRS allows exporting data from OBS to HDFS for computing and analyzing. After the analyzing and computing are complete, you can store the data in HDFS or export them to OBS. HDFS and OBS can also store the compressed data in the format of **bz2** or **gz**.

Submitting a Job on the GUI

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. For details, see **Synchronizing IAM Users to MRS**.

NOTE

- When the policy of the user group to which the IAM user belongs changes from MRS ReadOnlyAccess to MRS CommonOperations, MRS FullAccess, or MRS Administrator, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from MRS CommonOperations, MRS FullAccess, or MRS Administrator to MRS ReadOnlyAccess, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

Step 4 Click the **Jobs** tab.

Step 5 Click **Create**. The **Create Job** page is displayed.

NOTE

If the IAM username contains spaces (for example, **admin 01**), a job cannot be created.

Step 6 In **Type**, select **MapReduce**. Configure other job information.

Table 2-1 Job configuration information

Parameter	Description
Name	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** You are advised to set different names for different jobs.
Program Path	Path of the program package to be executed. The following requirements must be met: • Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. • The path of the program to be executed can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **obs://**. Example: **obs://wordcount/program/*xxx*.jar** – HDFS: The path must start with **/user**. For details about how to import data to HDFS, see **Importing Data**. • For SparkScript and HiveScript, the path must end with **.sql**. For MapReduce, the path must end with **.jar**. For Flink and SparkSubmit, the path must end with **.jar** or **.py**. The **.sql**, **.jar**, and **.py** are case-insensitive.
Parameters	(Optional) It is the key parameter for program execution. Multiple parameters are separated by space. Configuration method: *Program class name Data input path Data output path* • Program class name: It is specified by a function in your program. MRS is responsible for transferring parameters only. • Data input path: Click **HDFS** or **OBS** to select a path or manually enter a correct path. • Data output path: Enter a directory that does not exist. The parameter contains a maximum of 150,000 characters. It cannot contain special characters ;|&><'$, but can be left blank. **CAUTION** If you enter a parameter with sensitive information (such as the login password), the parameter may be exposed in the job details display and log printing. Exercise caution when performing this operation.
Service Parameter	(Optional) It is used to modify service parameters for the job. The parameter modification applies only to the current job. To make the modification take effect permanently for the cluster, follow instructions in **Configuring Service Parameters**. To add multiple parameters, click ⊕ on the right. To delete a parameter, click **Delete** on the right. **Table 2-2** lists the common service configuration parameters.

Parameter	Description
Command Reference	Command submitted to the background for execution when a job is submitted.

Table 2-2 Service Parameter parameters

Parameter	Description	Example Value
fs.obs.access.key	Key ID for accessing OBS.	-
fs.obs.secret.key	Key corresponding to the key ID for accessing OBS.	-

Step 7 Confirm job configuration information and click **OK**.

After the job is created, you can manage it.

----End

Submitting a Job in the Background

The default client installation path for MRS 3.x or later is /opt/Bigdata/client, and for versions earlier than MRS 3.x is /opt/client. Configure the path based on site requirements.

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 On the **Nodes** tab page, click the name of a Master node to go to the ECS management console.

Step 4 Click **Remote Login** in the upper right corner of the page.

Step 5 Enter the username and password of the Master node as prompted. The username is **root** and the password is the one set during cluster creation.

Step 6 Run the following command to initialize environment variables:

source /opt/Bigdata/client/bigdata_env

Step 7 If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If the Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 8 Run the following command to copy the program in the OBS file system to the Master node in the cluster:

hadoop fs -Dfs.obs.access.key=AK -Dfs.obs.secret.key=SK -copyToLocal source_path.jar target_path.jar

Example: **hadoop fs -Dfs.obs.access.key=XXXX -Dfs.obs.secret.key=XXXX -copyToLocal "obs://mrs-word/program/hadoop-mapreduce-examples-XXX.jar" "/home/omm/hadoop-mapreduce-examples-XXX.jar"**

You can log in to OBS Console using AK/SK. To obtain AK/SK information, click the username in the upper right corner of the management console and choose **My Credentials** > **Access Keys**.

Step 9 Run the following command to submit a wordcount job. If data needs to be read from OBS or outputted to OBS, the AK/SK parameters need to be added.

source /opt/Bigdata/client/bigdata_env;hadoop jar execute_jar wordcount input_path output_path

Example: **source /opt/Bigdata/client/bigdata_env;hadoop jar /home/omm/hadoop-mapreduce-examples-XXX.jar wordcount -Dfs.obs.access.key=XXXX -Dfs.obs.secret.key=XXXX "obs://mrs-word/input/*" "obs://mrs-word/output/"**

In the preceding command, **input_path** indicates a path for storing job input files on OBS. **output_path** indicates a path for storing job output files on OBS and needs to be set to a directory that does not exist

----End

2.5 Using Clusters with Kerberos Authentication Enabled

This section instructs you to use security clusters and run MapReduce, Spark, and Hive programs.

The Presto component of MRS 3.x does not support Kerberos authentication.

You can get started by reading the following topics:

1. **Creating a Security Cluster and Logging In to Manager**
2. **Creating a Role and a User**
3. **Running a MapReduce Program**
4. **Running a Spark Program**
5. **Running a Hive Program**

Creating a Security Cluster and Logging In to Manager

Step 1 Create a security cluster. For details, see **Creating a Custom Cluster**. Enable **Kerberos Authentication**, set **Password**, and confirm the password. This password is used to log in to Manager. Keep it secure.

Step 2 Log in to the MRS console.

Step 3 In the navigation pane on the left, choose **Active Clusters** and click the target cluster name on the right to access the cluster details page.

Step 4 Click **Access Manager** on the right of **MRS Manager** to log in to Manager.

- If you have bound an EIP when creating the cluster, perform the following operations:
  a. Add a security group rule. By default, your public IP address used for accessing port 9022 is filled in the rule. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

     NOTE

     - It is normal that the automatically generated public IP address is different from your local IP address and no action is required.
     - If port 9022 is a Knox port, you need to enable the permission to access port 9022 of Knox for accessing Manager.

  b. Select **I confirm that xx.xx.xx.xx is a trusted public IP address and MRS Manager can be accessed using this IP address.**
- If you have not bound an EIP when creating the cluster, perform the following operations:
  a. Select an available EIP from the drop-down list or click **Manage EIP** to create one.
  b. Add a security group rule. By default, your public IP address used for accessing port 9022 is filled in the rule. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

     NOTE

     - It is normal that the automatically generated public IP address is different from the local IP address and no action is required.
     - If port 9022 is a Knox port, you need to enable the permission of port 9022 to access Knox for accessing MRS Manager.

  c. Select **I confirm that xx.xx.xx.xx is a trusted public IP address and MRS Manager can be accessed using this IP address.**

Step 5 Click **OK**. The Manager login page is displayed. To assign permissions to other users to access Manager, add their public IP addresses as trusted ones by referring to **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 6 Enter the default username **admin** and the password you set when creating the cluster, and click **Log In**.

----End

Creating a Role and a User

For clusters with Kerberos authentication enabled, perform the following steps to create a user and assign permissions to the user to run programs.

Step 1 On Manager, choose **System** > **Permission** > **Role**.

Step 2 Click **Create Role**. For details, see **Creating a Role**.

Specify the following information:

- Enter a role name, for example, **mrrole**.
- In **Configure Resource Permission**, select the cluster to be operated, choose **Yarn** > **Scheduler Queue** > **root**, and select **Submit** and **Admin** in the **Permission** column. After you finish configuration, do not click **OK** but click the name of the target cluster shown in the following figure and then configure other permissions.
- Choose **HBase** > **HBase Scope**. Locate the row that contains **global**, and select **create**, **read**, **write**, and **execute** in the **Permission** column. After you finish configuration, do not click **OK** but click the name of the target cluster shown in the following figure and then configure other permissions.
- Choose **HDFS** > **File System** > **hdfs://hacluster/** and select **Read**, **Write**, and **Execute** in the **Permission** column. After you finish configuration, do not click **OK** but click the name of the target cluster shown in the following figure and then configure other permissions.
- Choose **Hive** > **Hive Read Write Privileges**, select **Select**, **Delete**, **Insert**, and **Create** in the **Permission** column, and click **OK**.

Step 3 Choose **System**. In the navigation pane on the left, choose **Permission** > **User Group** > **Create User Group** to create a user group for the sample project, for example, **mrgroup**. For details, see **Creating a User Group**.

Step 4 Choose **System**. In the navigation pane on the left, choose **Permission** > **User** > **Create** to create a user for the sample project. For details, see **Creating a User**.

- Enter a username, for example, **test**. If you want to run a Hive program, enter **hiveuser** in **Username**.
- Set **User Type** to **Human-Machine**.
- Enter a password. This password will be used when you run the program.
- In **User Group**, add **mrgroup** and **supergroup**.
- Set **Primary Group** to **supergroup** and bind the **mrrole** role to obtain the permission.

  Click **OK**.

Step 5 Choose **System**. In the navigation pane on the left, choose **Permission** > **User**, locate the row where user **test** locates, and select **Download Authentication Credential** from the **More** drop-down list. Save the downloaded package and decompress it to obtain the **keytab** and **krb5.conf** files.

----End

Running a MapReduce Program

This section describes how to run a MapReduce program in security cluster mode.

Prerequisites

You have compiled the program and prepared data files, for example, **mapreduce-examples-1.0.jar**, **input_data1.txt**, and **input_data2.txt**..

Procedure

Step 1 Use a remote login software (for example, MobaXterm) to log in to the master node of the security cluster using SSH (using the EIP).

Step 2 After the login is successful, run the following commands to create the **test** folder in the **/opt/Bigdata/client** directory and create the **conf** folder in the **test** directory:

```
cd /opt/Bigdata/client
mkdir test
cd test
mkdir conf
```

Step 3 Use an upload tool (for example, WinSCP) to copy **mapreduce-examples-1.0.jar**, **input_data1.txt**, and **input_data2.txt** to the **test** directory, and copy the **keytab** and **krb5.conf** files obtained in **Step 5** in **Creating Roles and Users** to the **conf** directory.

Step 4 Run the following commands to configure environment variables and authenticate the created user, for example, **test**:

```
cd /opt/Bigdata/client
source bigdata_env
export YARN_USER_CLASSPATH=/opt/Bigdata/client/test/conf/
kinit test
```

Enter the password as prompted. If no error message is displayed (you need to change the password as prompted upon the first login), Kerberos authentication is complete.

Step 5 Run the following commands to import data to the HDFS:

```
cd test
hdfs dfs -mkdir /tmp/input
hdfs dfs -put input_data* /tmp/input
```

Step 6 Run the following commands to run the program:

```
yarn jar mapreduce-examples-1.0.jar com.xxx.bigdata.mapreduce.examples.FemaleInfoCollector /tmp/
input /tmp/mapreduce_output
```

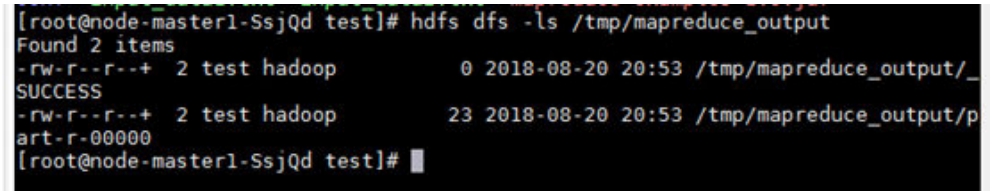
In the preceding commands:

/tmp/input indicates the input path in the HDFS.

/tmp/mapreduce_output indicates the output path in the HDFS. This directory must not exist. Otherwise, an error will be reported.

Step 7 After the program is executed successfully, run the **hdfs dfs -ls /tmp/mapreduce_output** command. The following command output is displayed.

Figure 2-1 Program running result



----End

Running a Spark Program

This section describes how to run a Spark program in security cluster mode.

Prerequisites

You have compiled the program and prepared data files, for example, **FemaleInfoCollection.jar**, **input_data1.txt**, and **input_data2.txt**..

Procedure

Step 1 Use a remote login software (for example, MobaXterm) to log in to the master node of the security cluster using SSH (using the EIP).

Step 2 After the login is successful, run the following commands to create the **test** folder in the **/opt/Bigdata/client** directory and create the **conf** folder in the **test** directory:

```
cd /opt/Bigdata/client
mkdir test
cd test
mkdir conf
```

Step 3 Use an upload tool (for example, WinSCP) to copy **FemaleInfoCollection.jar**, **input_data1.txt**, and **input_data2.txt** to the **test** directory, and copy the **keytab** and **krb5.conf** files obtained in **Step 5** in section **Creating Roles and Users** to the **conf** directory.

Step 4 Run the following commands to configure environment variables and authenticate the created user, for example, **test**:

```
cd /opt/Bigdata/client
source bigdata_env
export YARN_USER_CLASSPATH=/opt/Bigdata/client/test/conf/
kinit test
```

Enter the password as prompted. If no error message is displayed, Kerberos authentication is complete.

Step 5 Run the following commands to import data to the HDFS:

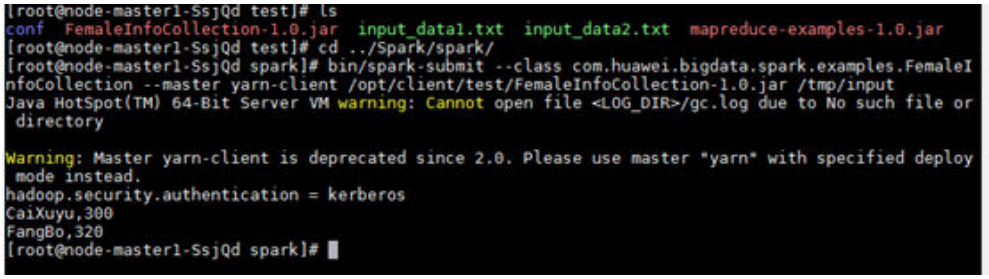
```
cd test
hdfs dfs -mkdir /tmp/input
hdfs dfs -put input_data* /tmp/input
```

Step 6 Run the following commands to run the program:

```
cd /opt/Bigdata/client/Spark/spark
bin/spark-submit --class com.xxx.bigdata.spark.examples.FemaleInfoCollection --master yarn-client /opt/Bigdata/client/test/FemaleInfoCollection-1.0.jar /tmp/input
```

Step 7 After the program is run successfully, the following information is displayed.

Figure 2-2 Program running result



----End

Running a Hive Program

This section describes how to run a Hive program in security cluster mode.

Prerequisites

You have compiled the program and prepared data files, for example, **hive-examples-1.0.jar**, **input_data1.txt**, and **input_data2.txt**..

Procedure

Step 1 Use a remote login software (for example, MobaXterm) to log in to the master node of the security cluster using SSH (using the EIP).

Step 2 After the login is successful, run the following commands to create the **test** folder in the **/opt/Bigdata/client** directory and create the **conf** folder in the **test** directory:

```
cd /opt/Bigdata/client
mkdir test
cd test
mkdir conf
```

Step 3 Use an upload tool (for example, WinSCP) to copy **FemaleInfoCollection.jar**, **input_data1.txt**, and **input_data2.txt** to the **test** directory, and copy the **keytab** and **krb5.conf** files obtained in **Step 5** in section **Creating Roles and Users** to the **conf** directory.

Step 4 Run the following commands to configure environment variables and authenticate the created user, for example, **test**:

```
cd /opt/Bigdata/client
source bigdata_env
export YARN_USER_CLASSPATH=/opt/Bigdata/client/test/conf/
kinit test
```

Enter the password as prompted. If no error message is displayed, Kerberos authentication is complete.

Step 5 Run the following command to run the program:

```
chmod +x /opt/hive_examples -R  cd /opt/hive_examples  java -cp .:hive-examples-1.0.jar:/opt/hive_examples/conf:/opt/Bigdata/client/Hive/Beeline/lib/*:/opt/Bigdata/client/HDFS/hadoop/lib/* com.xxx.bigdata.hive.example.ExampleMain
```

Step 6 After the program is run successfully, the following information is displayed.

Figure 2-3 Program running result

----End

2.6 Terminating a Cluster

You can terminate an MRS cluster that is no longer use after job execution is complete.

Background

You can manually delete a cluster after data analysis is complete or when the cluster encounters an exception. A cluster failed to be deployed will be automatically deleted.

Procedure

Step 1 Log in to the MRS management console.

Step 2 In the navigation pane on the left, choose **Active Clusters**.

Step 3 In the cluster list, locate the row containing the cluster to be deleted, and click **Delete** in the **Operation** column.

The cluster status changes from **Running** to **Deleting**, and finally to **Deleted**. You can view the deleted cluster in **Cluster History**.

----End

3 Preparing a User

3.1 Creating an MRS User

Use IAM to implement fine-grained permission control over your MRS. With IAM, you can:

- Create IAM users under your cloud account for employees based on your enterprise's organizational structure so that each employee is allowed to access MRS resources using their unique security credential (IAM user).
- Grant only the permissions required for users to perform a specific task.
- Entrust a cloud account or cloud service to perform efficient O&M on your MRS resources.

If your cloud account does not require IAM users, skip this section.

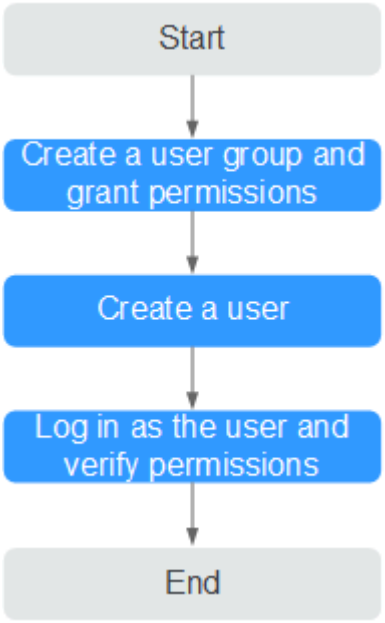
This section describes the procedure for granting permissions (see **Figure 3-1**).

Prerequisites

Learn about the permissions.

Process Flow

Figure 3-1 Process for granting MRS permissions



1. Create a user group and assign permissions to it.

   Create a user group on the IAM console, and assign MRS permissions to the group.

2. Create a user and add it to a user group.

   Create a user on the IAM console and add the user to the group created in **1. Create a user group and assign permissions to it**.

3. Log in and verify permissions.

   Log in to the console by using the user created, and verify that the user has the granted permissions.

   - Choose **Service List** > **MapReduce Service**. Then click **Create Cluster** on the MRS console. If a message appears indicating that you have insufficient permissions to perform the operation, the **MRS ReadOnlyAccess** policy has already taken effect.
   - Choose any other service in **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the **MRS ReadOnlyAccess** policy has already taken effect.

MRS Permission Description

By default, new IAM users do not have any permissions. To assign permissions to a user, add the user to one or more groups and assign permissions policies or roles to these groups. The user then inherits permissions from the groups it is a member of and can perform specified operations on cloud services based on the permissions.

MRS is a project-level service deployed and accessed in specific physical regions. To assign permissions to a user group, specify **Scope** as **Region-specific projects** and select projects in the corresponding region for the permissions to take effect.

If **All projects** is selected, the permissions will take effect for the user group in all region-specific projects. When accessing MRS, the users need to switch to a region where they have been authorized to use the MRS service.

You can grant users permissions by using roles and policies.

- Roles: A type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. This mechanism provides only a limited number of service-level roles for authorization. When using roles to grant permissions, you need to also assign other roles on which the permissions depend to take effect. However, roles are not an ideal choice for fine-grained authorization and secure access control.
- Policies: A type of fine-grained authorization mechanism that defines permissions required to perform operations on specific cloud resources under certain conditions. This mechanism allows for more flexible policy-based authorization, meeting requirements for secure access control. For example, you can grant MRS users only the permissions for performing specified operations on MRS clusters, such as creating a cluster and querying a cluster list rather than deleting a cluster. Most policies define permissions based on APIs.

Table 3-1 lists all the system policies supported by MRS.

Table 3-1 MRS system policies

Policy	Description	Type
MRS FullAccess	Administrator permissions for MRS. Users granted these permissions can operate and use all MRS resources.	Fine-grained policy
MRS CommonOperations	Common user permissions for MRS. Users granted these permissions can use MRS but cannot add or delete resources.	Fine-grained policy
MRS ReadOnlyAccess	Read-only permission for MRS. Users granted these permissions can only view MRS resources.	Fine-grained policy
MRS Administrator	Permissions: • All operations on MRS • Users with permissions of this policy must also be granted permissions of the **Tenant Guest**, **Server Administrator**, and **BSS Administrator** policies.	RBAC policy

Table 3-2 lists the common operations supported by each system-defined policy or role of MRS. Select the policies or roles as required.

Table 3-2 Common operations supported by each system-defined policy

Operation	MRS FullAccess	MRS CommonOperations	MRS ReadOnlyAccess	MRS Administrator
Creating a cluster	√	x	x	√
Resizing a cluster	√	x	x	√
Upgrading node specifications	√	x	x	√
Deleting a cluster	√	x	x	√
Querying cluster details	√	√	√	√
Querying a cluster list	√	√	√	√
Configuring an auto scaling rule	√	x	x	√
Querying a host list	√	√	√	√
Querying operation logs	√	√	√	√
Creating and executing a job	√	√	x	√
Stopping a job	√	√	x	√
Deleting a single job	√	√	x	√
Deleting jobs in batches	√	√	x	√
Querying job details	√	√	√	√

Operation	MRS FullAccess	MRS CommonOperations	MRS ReadOnlyAccess	MRS Administrator
Querying a job list	√	√	√	√
Creating a folder	√	√	x	√
Deleting a file	√	√	x	√
Querying a file list	√	√	√	√
Operating cluster tags in batches	√	√	x	√
Creating a single cluster tag	√	√	x	√
Deleting a single cluster tag	√	√	x	√
Querying a resource list by tag	√	√	√	√
Querying cluster tags	√	√	√	√
Accessing Manager	√	√	x	√
Querying a patch list	√	√	√	√
Installing a patch	√	√	x	√
Uninstalling a patch	√	√	x	√
Authorizing O&M channels	√	√	x	√
Sharing O&M channel logs	√	√	x	√

Operation	MRS FullAccess	MRS CommonOperations	MRS ReadOnlyAccess	MRS Administrator
Querying an alarm list	√	√	√	√
Subscribing to alarm notification	√	√	x	√
Submitting an SQL statement	√	√	x	√
Querying SQL results	√	√	x	√
Canceling an SQL execution task	√	√	x	√

3.2 Creating a Custom Policy

Custom policies can be created to supplement the system-defined policies of MRS.

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Edit JSON policies from scratch or based on an existing policy.

Example Custom Policies

- Example 1: Allowing users to create MRS clusters only

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "mrs:cluster:create",
                "ecs:*:*",
                "bms:*:*",
                "evs:*:*",
                "vpc:*:*",
                "smn:*:*"
            ]
        }
    ]
}
```

- Example 2: Allowing users to resize an MRS cluster

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "mrs:cluster:resize"
         ]
      }
   ]
}
```

- Example 3: Allowing users to create a cluster, create and execute a job, and delete a single job, but denying cluster deletion

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "mrs:cluster:create",
            "mrs:job:submit",
            "mrs:job:delete"
         ]
      },
      {
         "Effect": "Deny",
         "Action": [
            "mrs:cluster:delete"
         ]
      }
   ]
}
```

- Example 4: Allowing users to create an ECS cluster with the minimum permission

  NOTE

  - If you need a key pair when creating a cluster, add the following permissions: **ecs:serverKeypairs:get** and **ecs:serverKeypairs:list**.
  - Add the **kms:cmk:list** permission when encrypting data disks during cluster creation.
  - Add the **mrs:alarm:subscribe** permission to enable the alarm function during cluster creation.
  - Add the **rds:instance:list** permission to use external data sources during cluster creation.

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "mrs:cluster:create"
         ]
      },
      {
         "Effect": "Allow",
         "Action": [
            "ecs:cloudServers:updateMetadata",
            "ecs:cloudServerFlavors:get",
            "ecs:cloudServerQuotas:get",
            "ecs:servers:list",
            "ecs:servers:get",
            "ecs:cloudServers:delete",
            "ecs:cloudServers:list",
            "ecs:serverInterfaces:get",
```

```
                "ecs:serverGroups:manage",
                "ecs:servers:setMetadata",
                "ecs:cloudServers:get",
                "ecs:cloudServers:create"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "vpc:securityGroups:create",
                "vpc:securityGroupRules:delete",
                "vpc:vpcs:create",
                "vpc:ports:create",
                "vpc:securityGroups:get",
                "vpc:subnets:create",
                "vpc:privateIps:delete",
                "vpc:quotas:list",
                "vpc:networks:get",
                "vpc:publicIps:list",
                "vpc:securityGroups:delete",
                "vpc:securityGroupRules:create",
                "vpc:privateIps:create",
                "vpc:ports:get",
                "vpc:ports:delete",
                "vpc:publicIps:update",
                "vpc:subnets:get",
                "vpc:publicIps:get",
                "vpc:ports:update",
                "vpc:vpcs:list"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "evs:quotas:get",
                "evs:types:get"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "bms:serverFlavors:get"
            ]
        }
    ]
}
```

- Example 5: Allowing users to create a BMS cluster with the minimum permission

  NOTE

  - If you need a key pair when creating a cluster, add the following permissions: **ecs:serverKeypairs:get** and **ecs:serverKeypairs:list**.
  - Add the **kms:cmk:list** permission when encrypting data disks during cluster creation.
  - Add the **mrs:alarm:subscribe** permission to enable the alarm function during cluster creation.
  - Add the **rds:instance:list** permission to use external data sources during cluster creation.

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "mrs:cluster:create"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ecs:servers:list",
                "ecs:servers:get",
                "ecs:cloudServers:delete",
                "ecs:serverInterfaces:get",
                "ecs:serverGroups:manage",
                "ecs:servers:setMetadata",
                "ecs:cloudServers:create",
                "ecs:cloudServerFlavors:get",
                "ecs:cloudServerQuotas:get"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "vpc:securityGroups:create",
                "vpc:securityGroupRules:delete",
                "vpc:vpcs:create",
                "vpc:ports:create",
                "vpc:securityGroups:get",
                "vpc:subnets:create",
                "vpc:privateIps:delete",
                "vpc:quotas:list",
                "vpc:networks:get",
                "vpc:publicIps:list",
                "vpc:securityGroups:delete",
                "vpc:securityGroupRules:create",
                "vpc:privateIps:create",
                "vpc:ports:get",
                "vpc:ports:delete",
                "vpc:publicIps:update",
                "vpc:subnets:get",
                "vpc:publicIps:get",
                "vpc:ports:update",
                "vpc:vpcs:list"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "evs:quotas:get",
                "evs:types:get"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "bms:servers:get",
                "bms:servers:list",
                "bms:serverQuotas:get",
                "bms:servers:updateMetadata",
                "bms:serverFlavors:get"
            ]
        }
    ]
}
```

- Example 6: Allowing users to create a hybrid ECS and BMS cluster with the minimum permission

NOTE

- If you need a key pair when creating a cluster, add the following permissions: **ecs:serverKeypairs:get** and **ecs:serverKeypairs:list**.
- Add the **kms:cmk:list** permission when encrypting data disks during cluster creation.
- Add the **mrs:alarm:subscribe** permission to enable the alarm function during cluster creation.
- Add the **rds:instance:list** permission to use external data sources during cluster creation.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "mrs:cluster:create"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ecs:cloudServers:updateMetadata",
        "ecs:cloudServerFlavors:get",
        "ecs:cloudServerQuotas:get",
        "ecs:servers:list",
        "ecs:servers:get",
        "ecs:cloudServers:delete",
        "ecs:cloudServers:list",
        "ecs:serverInterfaces:get",
        "ecs:serverGroups:manage",
        "ecs:servers:setMetadata",
        "ecs:cloudServers:get",
        "ecs:cloudServers:create"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "vpc:securityGroups:create",
        "vpc:securityGroupRules:delete",
        "vpc:vpcs:create",
        "vpc:ports:create",
        "vpc:securityGroups:get",
        "vpc:subnets:create",
        "vpc:privateIps:delete",
        "vpc:quotas:list",
        "vpc:networks:get",
        "vpc:publicIps:list",
        "vpc:securityGroups:delete",
        "vpc:securityGroupRules:create",
        "vpc:privateIps:create",
        "vpc:ports:get",
        "vpc:ports:delete",
        "vpc:publicIps:update",
        "vpc:subnets:get",
        "vpc:publicIps:get",
        "vpc:ports:update",
        "vpc:vpcs:list"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "evs:quotas:get",
        "evs:types:get"
      ]
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "bms:servers:get",
                "bms:servers:list",
                "bms:serverQuotas:get",
                "bms:servers:updateMetadata",
                "bms:serverFlavors:get"
            ]
        }
    ]
}
```

3.3 Synchronizing IAM Users to MRS

IAM user synchronization is to synchronize IAM users bound with MRS policies to the MRS system and create accounts with the same usernames but different passwords as the IAM users. Then, you can use an IAM username (the password needs to be reset by user **admin** of Manager) to log in to Manager for cluster management, and submit jobs on the GUI in a cluster with Kerberos authentication enabled.

Table 3-3 compares IAM users' permission policies and the synchronized users' permissions on MRS. For details about the default permissions on Manager, see **Users and Permissions of MRS Clusters**.

Table 3-3 Policy and permission mapping after synchronization

Policy Type	IAM Policy	User's Default Permissions on MRS After Synchronization	Have Permission to Perform the Synchronization	Have Permission to Submit Jobs
Fine-grained	MRS ReadOnlyAccess	Manager_viewer	No	No
	MRS CommonOperations	• Manager_viewer • default • launcher-job	No	Yes

Policy Type	IAM Policy	User's Default Permissions on MRS After Synchronization	Have Permission to Perform the Synchronization	Have Permission to Submit Jobs
	MRS FullAccess	• Manager_administrator • Manager_auditor • Manager_operator • Manager_tenant • Manager_viewer • System_administrator • default • launcher-job	Yes	Yes
RBAC	MRS Administrator	• Manager_administrator • Manager_auditor • Manager_operator • Manager_tenant • Manager_viewer • System_administrator • default • launcher-job	No	Yes

Policy Type	IAM Policy	User's Default Permissions on MRS After Synchronization	Have Permission to Perform the Synchronization	Have Permission to Submit Jobs
	Server Administrator, Tenant Guest, and MRS Administrator	• Manager_administrator • Manager_auditor • Manager_operator • Manager_tenant • Manager_viewer • System_administrator • default • launcher-job	Yes	Yes
	Tenant Administrator	• Manager_administrator • Manager_auditor • Manager_operator • Manager_tenant • Manager_viewer • System_administrator • default • launcher-job	Yes	Yes

Policy Type	IAM Policy	User's Default Permissions on MRS After Synchronization	Have Permission to Perform the Synchronization	Have Permission to Submit Jobs
Custom	Custom policy	• Manager_viewer • default • launcher-job	• If custom policies use RBAC policies as a template, refer to the RBAC policies. • If custom policies use fine-grained policies as a template, refer to the fine-grained policies. The fine-grained policies are recommended.	Yes

NOTE

To facilitate user permission management, use fine-grained policies rather than RBAC policies. In fine-grained policies, the Deny action takes precedence over other actions.

- A user has permission to synchronize IAM users only when the user has the Tenant Administrator role or has the Server Administrator, Tenant Guest, and MRS Administrator roles at the same time.
- A user with the **action:mrs:cluster:syncUser** policy has permission to synchronize IAM users.

Procedure

Step 1 Create a user and authorize the user to use MRS. For details, see **Creating an MRS User**.

Step 2 Log in to the MRS management console and create a cluster. For details, see **Creating a Custom Cluster**.

Step 3 In the left navigation pane, choose **Clusters** > **Active Clusters**. Click the cluster name to go to the cluster details page.

Step 4 In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

Step 5 After a synchronization request is sent, choose **Operation Logs** in the left navigation pane on the MRS console to check whether the synchronization is successful. For details about the logs, see **Viewing MRS Operation Logs**.

Step 6 After the synchronization is successful, use the user synchronized with IAM to perform subsequent operations.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.
- After you click **Synchronize** on the right side of **IAM User Sync**, the cluster details page is blank for a short time, because user data is being synchronized. The page will be properly displayed after the data synchronization is complete.

- Submitting jobs in a security cluster: Users can submit jobs using the job management function on the GUI in the security cluster. For details, see **Running a MapReduce Job**.
- All tabs are displayed on the cluster details page, including **Components**, **Tenants**, and **Backups & Restorations**.
- Logging in to Manager

  a. Log in to Manager as user **admin**. For details, see **Accessing Manager**.

  b. Initialize the password of the user synchronized with IAM. For details, see **Initializing the Password of a System User**.

  c. Modify the role bound to the user group to which the user belongs to control user permissions on Manager. For details, see **Related Tasks**. For details about how to create and modify a role, see **Creating a Role**. After the component role bound to the user group to which the user belongs is modified, it takes some time for the role permissions to take effect.

  d. Log in to Manager using the user synchronized with IAM and the password after the initialization in **Step 6.b**.

NOTE

If the IAM user's permission changes, go to **Step 4** to perform second synchronization. After the second synchronization, a system user's permissions are the union of the permissions defined in the IAM system policy and the permissions of roles added by the system user on Manager. After the second synchronization, a custom user's permissions are subject to the permissions configured on Manager.

- System user: If all user groups to which an IAM user belongs are bound to system policies (RABC policies and fine-grained policies belong to system policies), the IAM user is a system user.
- Custom user: If the user group to which an IAM user belongs is bound to any custom policy, the IAM user is a custom user.

----End

4 Configuring a Cluster

4.1 Methods of Creating MRS Clusters

This section describes how to MRS clusters.

- **Quick Creation of a Hadoop Analysis Cluster**: On the **Quick Config** tab page, you can quickly configure parameters to Hadoop analysis clusters within a few minutes, facilitating analysis and queries of vast amounts of data.
- **Quick Creation of an HBase Analysis Cluster**: On the **Quick Config** tab page, you can quickly configure parameters to HBase query clusters within a few minutes, facilitating storage and distributed computing of vast amounts of data.
- **Quick Creation of a Kafka Streaming Cluster**: On the **Quick Config** tab page, you can quickly configure parameters to Kafka streaming clusters within a few minutes, facilitating streaming data ingestion as well as real-time data processing and storage.
- **Quick Creation of a ClickHouse Cluster**: You can quickly a ClickHouse cluster. ClickHouse is a columnar database management system used for online analysis. It features the ultimate compression rate and fast query performance.
- **Quick Creation of a Real-time Analysis Cluster**: You can a real-time analysis cluster within a few minutes to quickly collect, analyze, and query a large amount of data.
- **Creating a Custom Cluster**: On the **Custom Config** tab page, you can flexibly configure parameters to clusters based on application scenarios, such as ECS specifications to better suit your service requirements.

4.2 Quick Creation of a Cluster

4.2.1 Quick Creation of a Hadoop Analysis Cluster

This section describes how to quickly a Hadoop analysis cluster for analyzing and querying vast amounts of data. In the open-source Hadoop ecosystem, Hadoop

uses Yarn to manage cluster resources, Hive and Spark to provide offline storage and computing of large-scale distributed data, Spark Streaming and Flink to offer streaming data computing, and Presto to enable interactive queries, Tez to provide a distributed computing framework of directed acyclic graphs (DAGs).

Quick Creation of a Hadoop Analysis Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Quick Config** tab.

Step 4 Configure basic cluster information. For details about the parameters, see **Creating a Custom Cluster**.

- **Region**: Use the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, for example, **mrs_20180321**.
- **Cluster Version**: Select the latest version, which is the default value. (The components provided by a cluster vary according to the cluster version. Select a cluster version based on site requirements.)
- **Component**: Select **Hadoop analysis cluster**.
- **AZ**: Use the default value.
- **VPC**: Use the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Use the default value.
- **Enterprise Project**: Use the default value.
- **CPU Architecture**: Use the default value.
- **Cluster Node**: Select the number of cluster nodes and node specifications based on site requirements. For MRS 3.*x* or later, the memory of the master node must be greater than 64 GB.
- **Cluster HA**: Use the default value. This parameter is not available in MRS 3.*x*.
- **Kerberos Authentication**: Select whether to enable Kerberos authentication.
- **Username**: The default value is **root/admin**. User **root** is used to remotely log in to ECSs, and user **admin** is used to access the cluster management page.
- **Password**: Set a password for user **root**/**admin**.
- **Confirm Password**: Enter the password of user **root**/**admin** again.

Step 5 Select **Enable** to enable secure communications. For details, see **Communication Security Authorization**.

Step 6 **Click Apply Now.**

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 7 Click **Back to Cluster List** to view the cluster status. Click **Access Cluster** to view cluster details.

For details about cluster status during creation, see the description of the status parameters in **Table 5-4**.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

On the MRS management console, a maximum of 10 clusters can be concurrently created, and a maximum of 100 clusters can be managed.

----End

4.2.2 Quick Creation of an HBase Analysis Cluster

This section describes how to quickly an HBase query cluster. The HBase cluster uses Hadoop and HBase components to provide a column-oriented distributed cloud storage system featuring enhanced reliability, excellent performance, and elastic scalability. It applies to the storage and distributed computing of massive amounts of data. You can use HBase to build a storage system capable of storing TB- or even PB-level data. With HBase, you can filter and analyze data with ease and get responses in milliseconds, rapidly mining data value.

Quick Creation of an HBase Analysis Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Quick Config** tab.

Step 4 Configure basic cluster information. For details about the parameters, see **Creating a Custom Cluster**.

- **Region**: Use the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, for example, **mrs_20180321**.
- **Cluster Version**: Select the latest version, which is the default value. (The components provided by a cluster vary according to the cluster version. Select a cluster version based on site requirements.)
- **Component**: Select **HBase Query Cluster**.
- **AZ**: Use the default value.
- **VPC**: Use the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Use the default value.
- **Enterprise Project**: Use the default value.
- **CPU Architecture**: Use the default value.
- **Cluster Node**: Select the number of cluster nodes and node specifications based on site requirements. For MRS 3.*x* or later, the memory of the master node must be greater than 64 GB.
- **Cluster HA**: Use the default value. This parameter is not available in MRS 3.*x*.
- **Kerberos Authentication**: Select whether to enable Kerberos authentication.

- **Username**: The default value is **root/admin**. User **root** is used to remotely log in to ECSs, and user **admin** is used to access the cluster management page.
- **Password**: Set a password for user **root**/**admin**.
- **Confirm Password**: Enter the password of user **root**/**admin** again.

Step 5 Select **Enable** to enable secure communications. For details, see **Communication Security Authorization**.

Step 6 **Click Apply Now.**

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 7 Click **Back to Cluster List** to view the cluster status. Click **Access Cluster** to view cluster details.

For details about cluster status during creation, see the description of the status parameters in **Table 5-4**.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

On the MRS management console, a maximum of 10 clusters can be concurrently created, and a maximum of 100 clusters can be managed.

----End

4.2.3 Quick Creation of a Kafka Streaming Cluster

This section describes how to quickly a Kafka streaming cluster. The Kafka cluster uses the Kafka and Storm components to provide an open-source messaging system with high throughput and scalability. It is widely used in scenarios such as log collection and monitoring data aggregation to implement efficient streaming data collection and real-time data processing and storage.

Quick Creation of a Kafka Streaming Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Quick Config** tab.

Step 4 Configure basic cluster information. For details about the parameters, see **Creating a Custom Cluster**.

- **Region**: Use the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, for example, **mrs_20200321**.
- **Cluster Version**: The components provided by a cluster vary according to the cluster version. Select a cluster version based on site requirements.
- **Component**: Select **Kafka streaming cluster**.

- **AZ**: Use the default value.
- **VPC**: Use the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Use the default value.
- **Enterprise Project**: Use the default value.
- **CPU Architecture**: Use the default value.
- **Cluster Node**: Select the number of cluster nodes and node specifications based on site requirements. For MRS 3.*x* or later, the memory of the master node must be greater than 64 GB.
- **Cluster HA**: Use the default value. This parameter is not available in MRS 3.*x*.
- LVM: Use the default value. This parameter is not available in MRS 3.*x*.
- **Kerberos Authentication**: Select whether to enable Kerberos authentication.
- **Username**: The default value is **root/admin**. User **root** is used to remotely log in to ECSs, and user **admin** is used to access the cluster management page.
- **Password**: Set a password for user **root**/**admin**.
- **Confirm Password**: Enter the password of user **root**/**admin** again.

Step 5 Select **Enable** to enable secure communications. For details, see **Communication Security Authorization**.

Step 6 **Click Apply Now.**

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 7 Click **Back to Cluster List** to view the cluster status. Click **Access Cluster** to view cluster details.

For details about cluster status during creation, see the description of the status parameters in **Table 5-4**.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

On the MRS management console, a maximum of 10 clusters can be concurrently created, and a maximum of 100 clusters can be managed.

----End

4.2.4 Quick Creation of a ClickHouse Cluster

This section describes how to quickly a ClickHouse cluster. ClickHouse is a columnar database management system used for online analysis. It features the ultimate compression rate and fast query performance. It is widely used in Internet advertisement, app and web traffic analysis, telecom, finance, and IoT fields.

The ClickHouse cluster consists of the following components:

- MRS 3.1.0: ClickHouse 21.3.4.25 and ZooKeeper 3.5.6.

The ClickHouse cluster table engine that uses Kunpeng as the CPU architecture does not support HDFS and Kafka.

Quick Creation of a ClickHouse Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Quick Config** tab.

Step 4 Configure basic cluster information. For details about the parameters, see **Creating a Custom Cluster**.

- **Region**: Use the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, Example: **mrs_20201121**.
- **Cluster Version**: Select the latest version, which is the default value. (The components provided by a cluster vary according to the cluster version. Select a cluster version based on site requirements.)
- **Component**: Select **ClickHouse cluster**.
- **AZ**: Use the default value.
- **VPC**: Use the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Use the default value.
- **Enterprise Project**: Use the default value.
- **CPU Architecture**: Use the default value.
- **Cluster Node**: Select the number of cluster nodes and node specifications based on site requirements. For MRS 3.*x* or later, the memory of the master node must be greater than 64 GB.
- **Kerberos Authentication**: Select whether to enable Kerberos authentication.
- **Username**: The default value is **root/admin**. User **root** is used to remotely log in to ECSs, and user **admin** is used to access the cluster management page.
- **Password**: Set a password for user **root**/**admin**.
- **Confirm Password**: Enter the password of user **root**/**admin** again.

Step 5 Select **Enable** to enable secure communications. For details, see **Communication Security Authorization**.

Step 6 **Click Apply Now.**

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 7 Click **Back to Cluster List** to view the cluster status. Click **Access Cluster** to view cluster details.

For details about cluster status during creation, see the description of the status parameters in **Table 5-4**.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

On the MRS management console, a maximum of 10 clusters can be concurrently created, and a maximum of 100 clusters can be managed.

----End

4.2.5 Quick Creation of a Real-time Analysis Cluster

This section describes how to quickly a real-time analysis cluster. The real-time analysis cluster uses Hadoop, Kafka, Flink, and ClickHouse to collect, analyze, and query a large amount of data in real time.

The real-time analysis cluster consists of the following components:

- MRS 3.1.0: Hadoop 3.1.1, Kafka 2.11-2.4.0, Flink 1.12.0, ClickHouse 21.3.4.25, ZooKeeper 3.5.6, and Ranger 2.0.0.

Quick Creation of a Real-time Analysis Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Quick Config** tab.

Step 4 Configure basic cluster information. For details about the parameters, see **Creating a Custom Cluster**.

- **Region**: Use the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, Example: **mrs_20201130**.
- **Cluster Version**: Select the latest version, which is the default value. (The components provided by a cluster vary according to the cluster version. Select a cluster version based on site requirements.)
- **Component**: Select **Real-time Analysis Cluster**.
- **AZ**: Use the default value.
- **VPC**: Use the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Use the default value.
- **Enterprise Project**: Use the default value.
- **CPU Architecture**: Use the default value.
- **Cluster Node**: Select the number of cluster nodes and node specifications based on site requirements. For MRS 3.*x* or later, the memory of the master node must be greater than 64 GB.
- **Kerberos Authentication**: Select whether to enable Kerberos authentication.
- **Username**: The default value is **root/admin**. User **root** is used to remotely log in to ECSs, and user **admin** is used to access the cluster management page.
- **Password**: Set a password for user **root**/**admin**.
- **Confirm Password**: Enter the password of user **root**/**admin** again.

Step 5 Select **Enable** to enable secure communications. For details, see **Communication Security Authorization**.

Step 6 **Click Apply Now.**

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 7 Click **Back to Cluster List** to view the cluster status. Click **Access Cluster** to view cluster details.

For details about cluster status during creation, see the description of the status parameters in **Table 5-4**.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

On the MRS management console, a maximum of 10 clusters can be concurrently created, and a maximum of 100 clusters can be managed.

----End

4.3 Creating a Custom Cluster

The first step of using MRS is to a cluster. This section describes how to create a cluster on the **Custom Config** tab of the MRS management console.

You can create an IAM user or user group on the IAM management console and grant it specific operation permissions, to perform refined resource management after registering an account. For details, see **Creating an MRS User**.

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

NOTE

When creating a cluster, pay attention to quota notification. If a resource quota is insufficient, increase the resource quota as prompted and create a cluster.

Step 3 Click the **Custom Config** tab.

Step 4 Configure cluster information by referring to **Software Configurations** and click **Next**.

Step 5 Configure cluster information by referring to **Hardware Configurations** and click **Next**.

Step 6 Set advanced options by referring to **(Optional) Advanced Configuration** and click **Apply Now**.

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 7 Click **Back to Cluster List** to view the cluster status.

For details about cluster status during creation, see the description of the status parameters in **Table 5-4**.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

On the MRS management console, a maximum of 10 clusters can be concurrently created, and a maximum of 100 clusters can be managed.

----End

Software Configurations

Table 4-1 MRS cluster software configuration

Parameter	Description
Region	Select a region. Cloud service products in different regions cannot communicate with each other over an intranet. For low network latency and quick access, select the nearest region.
Cluster Name	The cluster name must be unique. A cluster name can contain 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. The default name is **mrs**_*xxxx*. *xxxx* is a random collection of letters and digits.
Cluster Version	Currently, MRS 2.1.1 and MRS 3.0.5 are supported.
Cluster Type	The cluster types are as follows: • **Analysis cluster**: is used for offline data analysis and provides Hadoop components. • **Streaming cluster**: is used for streaming tasks and provides stream processing components. • **Hybrid cluster**: is used for both offline data analysis and streaming processing and provides Hadoop components and streaming processing components. You are advised to use a hybrid cluster to perform offline data analysis and streaming processing tasks at the same time. • **Custom**: You can adjust the cluster service deployment mode based on service requirements. For details, see **Creating a Custom Topology Cluster**. (**Custom** is supported in MRS 3.x only.) **NOTE** • MRS streaming clusters do not support job and file management functions. • To install all components in a cluster, select **Custom**.

Parameter	Description
Components	MRS components are as follows. Components of an analysis cluster: • Presto: open source and distributed SQL query engine • Hadoop: distributed system architecture • Spark: in-memory distributed computing framework (not supported in MRS 3.*x*) • Spark2x: A fast general-purpose engine for large-scale data processing. It is developed based on the open-source Spark2.x version. (supported only by MRS 3.*x*) • Hive: data warehouse framework built on Hadoop • HBase: distributed column-oriented database • Tez: an application framework which allows for a complex directed-acyclic-graph of tasks for processing data • Hue: provides the Hadoop UI capability, which enables users to analyze and process Hadoop cluster data on browsers • Loader: a tool based on source Sqoop 1.99.7, designed for efficiently transferring bulk data between Apache Hadoop and structured datastores such as relational databases (not supported in MRS 3.*x*) Hadoop is mandatory, and Spark and Hive must be used together. Select components based on service requirements. • Flink: a distributed big data processing engine that can perform stateful computations over both finite and infinite data streams • Oozie: a Hadoop job scheduling system (supported only by MRS 3.*x*) • Alluxio: a memory speed virtual distributed storage system • Ranger: a framework to enable, monitor, and manage data security across the Hadoop platform • Impala: an SQL query engine for processing huge volumes of data • ClickHouse: A column database management system (DBMS) for on-line analytical processing (OLAP). The ClickHouse cluster table engine that uses Kunpeng as the CPU architecture does not support HDFS and Kafka. • Kudu: a column-oriented data store Components of a streaming cluster:

Parameter	Description
	• Kafka: distributed messaging system • Flume: distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of log data
Metadata	Whether to use external data sources to store metadata. • **Local**: Metadata is stored in the local cluster. • **Data connections**: Metadata of external data sources is used. If the cluster is abnormal or deleted, metadata is not affected. This mode applies to scenarios where storage and compute are decoupled. Clusters that support the Hive or Ranger component support this function.
Component	This parameter is valid only when **Metadata** is set to **Data connections**. It indicates the type of an external data source. This function is not available in MRS 3.*x*. • Hive • Ranger
Data Connection Type	This parameter is valid only when **Metadata** is set to **Data connections**. It indicates the type of an external data source. • Hive supports the following data connection types: – RDS PostgreSQL database – RDS MySQL database – Local database • Ranger supports the following data connection types: – RDS MySQL database – Local database
Data Connection Instance	This parameter is valid only when **Data Connection Type** is set to **RDS PostgreSQL database** or **RDS MySQL database**. This parameter indicates the name of the connection between the MRS cluster and the RDS database. This instance must be created before being referenced here. You can click **Create Data Connection** to create a data connection. For details, see **Configuring Data Connections**.

Hardware Configurations

Table 4-2 MRS cluster hardware configuration

Parameter	Description
AZ	Select the AZ associated with the region of the cluster. An AZ is a physical area that uses independent power and network resources. AZs are physically isolated but interconnected through the internal network. This improves the availability of applications. You are advised to create clusters in different AZs.
VPC	A VPC is a secure, isolated, and logical network environment. Select the VPC for which you want to create a cluster and click **View VPC** to view the name and ID of the VPC. If no VPC is available, create one.
Subnet	A subnet provides dedicated network resources that are isolated from other networks, improving network security. Select the subnet for which you want to create a cluster. Click **View Subnet** to view details about the selected subnet. If no subnet is created in the VPC, go to the VPC console and choose **Subnets** > **Create Subnet** to create one. For details about how to configure network ACL outbound rules, see **How Do I Configure a Network ACL Outbound Rule?** **NOTE** The number of IP addresses required by creating an MRS cluster depends on the number of cluster nodes and selected components, but not the cluster type. In MRS, IP addresses are automatically assigned to clusters during cluster creation basically based on the following formula: Quantity of IP addresses = Number of cluster nodes + 2 (one for Manager; one for the DB). In addition, if the Hadoop, Hue, Sqoop, and Presto or Loader and Presto components are selected during cluster deployment, one IP address is added for each component. To a ClickHouse cluster independently, the number of IP addresses required is calculated as follows: Number of IP addresses = Number of cluster nodes + 1 (for Manager).

Parameter	Description
Security Group	A security group is a set of ECS access rules. It provides access policies for ECSs that have the same security protection requirements and are mutually trusted in a VPC. When you create a cluster, you can select **Auto create** from the drop-down list of **Security Group** to create a security group or select an existing security group. **NOTE** When you select a security group created by yourself, ensure that the inbound rule contains a rule in which **Protocol** is set to **All**, **Port** is set to **All**, and **Source** is set to a trusted accessible IP address range. Do not use **0.0.0.0/0** as a source address. Otherwise, security risks may occur. If you do not know the trusted accessible IP address range, select **Auto create**.
EIP	After binding an EIP to an MRS cluster, you can use the EIP to access the Manager web UI of the cluster. When creating a cluster, you can select an available EIP from the drop-down list and bind it. If no EIP is available in the drop-down list, click **Manage EIP** to access the **EIPs** service page to one. **NOTE** The EIP must be in the same region as the cluster.
Enterprise Project	Select the enterprise project to which the cluster belongs. To use an enterprise project, create one on the **Enterprise** > **Project Management** page. The **Enterprise Management** console of the enterprise project is designed for resource management. It helps enterprises manage cloud-based personnel, resources, permissions, and finance in a hierarchical manner, such as management of companies, departments, and projects.

Table 4-3 Cluster node information

Parameter	Description
CPU Architecture	CPU architecture supported by MRS: • **x86**: The x86-based CPU architecture uses Complex Instruction Set Computing (CISC). Each instruction can be used to execute low-level hardware operations. The number of instructions is large, and the length of each instruction is different. Therefore, executing such an instruction is complex and time-consuming. • **Kunpeng**: The Kunpeng-based CPU architecture uses Reduced Instruction Set Computing (RISC). RISC is a microprocessor that executes fewer types of computer instructions but at a higher speed than CISC. RISC simplifies the computer architecture and improves the running speed. Compared with the x86-based CPU architecture, the Kunpeng-based CPU architecture has a more balanced performance and power consumption ratio. Kunpeng features high density, low power consumption, high cost-effectiveness.
Common Template	This parameter is valid only when **Cluster Type** is set to **Custom**. For details, see **Custom Cluster Template Description**.

Parameter	Description
Node Type	MRS provides three types of nodes: ● Master: A Master node in an MRS cluster manages the cluster, assigns executable cluster files to Core nodes, traces the execution status of each job, and monitors the DataNode running status. ● Core: A Core node in a cluster processes data and stores process data in HDFS. Analysis Core nodes are created in an analysis cluster. Streaming Core nodes are created in a streaming cluster. Both analysis and streaming Core nodes are created in a hybrid cluster. ● Task: A Task node in a cluster is used for computing and does not store persistent data. Yarn and Storm are mainly installed on Task nodes. Task nodes are optional, and the number of Task nodes can be zero. Analysis Task nodes are created in an analysis cluster. Streaming Task nodes are created in a streaming cluster. Both analysis and streaming Task nodes are created in a hybrid cluster. When the data volume change is small in a cluster but the cluster's service processing capabilities need to be remarkably and temporarily improved, add Task nodes to address the following situations: – Service volumes temporarily increase, for example, report processing at the end of the year. – Long-term tasks must be completed in a short time, for example, some urgent analysis tasks.
Instance Specifications	Instance specifications of Master or Core nodes. MRS supports host specifications determined by CPU, memory, and disk space. Click  to configure the instance specifications, system disk, and data disk parameters of the cluster node. **NOTE** ● More advanced instance specifications provide better data processing. ● If you select non-HDD disks for Core nodes, the disk types of Master and Core nodes are determined by **Data Disk**. ● If **Sold out** appears next to an instance specification of a node, the node of this specification cannot be d. You can only nodes of other specifications. ● For MRS 3.*x* or later, the memory of the master node must be greater than 64 GB.

Parameter	Description
System Disk	Storage type and storage space of the system disk on a node. Storage type can be any of the following: ● SATA: common I/O ● SAS: high I/O ● SSD: ultra-high I/O ● GPSSD: general-purpose SSD
Data Disk	Data disk storage space of a node. To increase data storage capacity, you can add disks at the same time when creating a cluster. The following two application scenarios are involved. ● Data storage and computing are separated. Data is stored in OBS, which features low cost and unlimited storage capacity. The clusters can be terminated at any time in OBS. The computing performance is determined by OBS access performance and is lower than that of HDFS. This configuration is recommended if data computing is infrequent. ● Data storage and computing are not separated. Data is stored in HDFS, which features high cost, high computing performance, and limited storage capacity. Before terminating clusters, you must export and store the data. This configuration is recommended if data computing is frequent. The storage type can be any of the following: ● SATA: common I/O ● SAS: high I/O ● SSD: ultra-high I/O ● GPSSD: general-purpose SSD **NOTE** More nodes in a cluster require higher disk capacity of Master nodes. To ensure stable cluster running, set the disk capacity of the Master node to over 600 GB if the number of nodes is 300 and increase it to over 1 TB if the number of nodes reaches 500.

Parameter	Description
Instance Count	Number of Master and Core nodes. For Master nodes: • If **Cluster HA** is enabled, the number of Master nodes is fixed to **2**. • If **Cluster HA** is disabled, the number of Master nodes is fixed to **1**. At least one Core node must exist and the total number of Core and Task nodes cannot exceed 500. Task: Click to add a Task node. Click to modify the instance specifications and disk configuration of a Task node. Click to delete the added Task node. **NOTE** • A maximum of 500 Core nodes are supported by default. If more than 500 Core nodes are required, contact technical support. • A small number of nodes may cause clusters to run slowly while a large number of nodes may be unnecessarily costly. Set an appropriate value based on data to be processed.
LVM	This parameter is valid when a streaming Core node is created only. Click this parameter to enable or disable the disk LVM management function. This parameter is not available in MRS 3.*x* and later versions. If LVM is enabled, all disks on a node are mounted as logical volumes. This delivers more proper disk planning to avoid data skew, thereby improving system stability.
Topology Adjustment	If the deployment mode in the **Common Node** does not meet the requirements, set **Topology Adjustment** to **Enable** and adjust the instance deployment mode based on service requirements. For details, see **Topology Adjustment for a Custom Cluster**. This parameter is valid only when **Cluster Type** is set to **Custom**.

(Optional) Advanced Configuration

Table 4-4 MRS cluster advanced configuration topology

Parameter	Description
Tag	For details, see **Adding a Tag to a Cluster**.
Hostname Prefix	Enter the prefix for the computer hostname of an ECS in the cluster.

Parameter	Description
Auto Scaling	Auto scaling can be configured only after you specify Task node specifications in the **Configure Hardware** step. For details about how to configure Task node specifications, see **Configuring an Auto Scaling Rule**.
Bootstrap Action	For details, see **Adding a Bootstrap Action**. This parameter is not available in MRS 3.*x*.
Agency	By binding an agency, ECSs or BMSs can manage some of your resources. Determine whether to configure an agency based on the actual service scenario. For example, you can configure an agency of the ECS type to automatically obtain the AK/SK to access OBS. For details, see **Configuring a Storage-Compute Decoupled Cluster (Agency)**. The **MRS_ECS_DEFAULT_AGENCY** agency has the OBSOperateAccess permission of OBS and the CESFullAccess (for users who have enabled fine-grained policies), CES Administrator, and KMS Administrator permissions in the region where the cluster is located.
Metric Sharing	Monitoring metrics of big data components are collected. If a fault occurs when you use a cluster, share the monitoring metrics withtechnical support personnel for troubleshooting. This parameter is not available in MRS 3.*x*.
OBS Permission Control	Users who have enabled fine-grained permission control can use this function to grant permissions on different directories in OBS file systems to different MRS users. For details, see **Configuring Fine-Grained Permissions for MRS Multi-User Access to OBS**. This parameter is not available in MRS 3.*x*.
Data Disk Encryption	Whether to encrypt data in the data disk mounted to the cluster. This function is disabled by default. To use this function, you must have the Security Administrator and KMS Administrator permissions. This parameter is not available in MRS 3.*x*. Keys used by encrypted data disks are provided by the Key Management Service (KMS) of the Data Encryption Workshop (DEW), secure and convenient. Therefore, you do not need to establish and maintain the key management infrastructure. Click **Data Disk Encryption** to enable or disable the data disk encryption function.
Key ID	This parameter is displayed only when the **Data Disk Encryption** function is enabled. This parameter indicates the key ID corresponding to the selected key name. This parameter is not available in MRS 3.*x*.

Parameter	Description
Key Name	This parameter is mandatory when the **Data Disk Encryption** function is enabled. Select the name of the key used to encrypt the data disk. By default, the default master key named **evs/default** is selected. You can select another master key from the drop-down list. This parameter is not available in MRS 3.*x*. If disks are encrypted using a CMK, which is then disabled or scheduled for deletion, the disks can no longer be read from or written to, and data on these disks may never be restored. Exercise caution when performing this operation. Click **View Key List** to enter a page where you can create and manage keys.
Alarm	If the alarm function is enabled, the cluster maintenance personnel can be notified in a timely manner to locate faults when the cluster runs abnormally or the system is faulty.
Rule Name	Name of the rule for sending alarm messages. The value can contain only digits, letters, hyphens (-), and underscores (_).
Topic Name	Select an existing topic or click **Create Topic** to create a topic. To deliver messages published to a topic, you need to add a subscriber to the topic. For details, see **Adding Subscriptions to a Topic**. A topic serves as a message sending channel, where publishers and subscribers can interact with each other.
Kerberos Authentication	Whether to enable Kerberos authentication when logging in to Manager. •  : If **Kerberos Authentication** is disabled, common users can use all functions of an MRS cluster. You are advised to disable Kerberos authentication in single-user scenarios. •  : If **Kerberos Authentication** is enabled, common users cannot use the file and job management functions of an MRS cluster and cannot view cluster resource usage or the job records for Hadoop and Spark. To use more cluster functions, the users must contact the Manager administrator to assign more permissions. You are advised to enable Kerberos authentication in multi-user scenarios.
Username	Name of the administrator of Manager. **admin** is used by default.

Parameter	Description
Password	Password of the Manager administrator The following requirements must be met: ● Must contain 8 to 26 characters. ● Must contain at least four of the following: – Lowercase letters – Uppercase letters – Digits – Have at least one of the following special characters: !?,.: -_{} [ ]@ $% ^ + = / ● Cannot be the same as the username or the username spelled backwards. Password Strength: The colorbar in red, orange, and green indicates weak, medium, and strong password, respectively.
Confirm Password	Enter the password of the Manager administrator again.

Parameter	Description
Login Mode	• Password You can log in to ECS nodes using a password. A password must meet the following requirements: 1. Must be a string and 8 to 26 characters long. 2. The password must contain at least four types of the following characters: uppercase letters, lowercase letters, digits, and special characters (! ?,.: -_{} [ ]@ $% ^ + = /). 3. The password cannot be the username or the reverse username. • Key Pair Key pairs are used to log in to ECS nodes of the cluster. Select a key pair from the drop-down list. Select "I acknowledge that I have obtained private key file *SSHkey-xxx* and that without this file I will not be able to log in to my ECS." If you have never created a key pair, click **View Key Pair** to create or import a key pair. And then, obtain a private key file. A key pair, also called an SSH key, consists of a public key and a private key. You can create an SSH key and download the private key for authenticating remote login. For security, a private key can only be downloaded once. Keep it secure. Use an SSH key in either of the following two methods: 1. Creating an SSH key: After you create an SSH key, a public key and a private key are generated. The public key is stored in the system, and the private key is stored in the local ECS. When you log in to an ECS, the public and private keys are used for authentication. 2. Importing an SSH key: If you have obtained the public and private keys, import the public key into the system. When you log in to an ECS, the public and private keys are used for authentication.

Parameter	Description
Secure Communications	MRS clusters provision, manage, and use big data components through the management console. Big data components are deployed in a user's VPC. If the MRS management console needs to directly access big data components deployed in the user's VPC, you need to enable the corresponding security group rules after you have obtained user authorization. This authorization process is called secure communications. For details, see **Communication Security Authorization**. If the secure communications function is not enabled, MRS clusters cannot be created.

Failed to Create a Cluster

If a cluster fails to be created, the failed task will be managed on the **Manage Failed Tasks** page. Choose **Clusters** > **Active Clusters**. Click shown in **Figure 4-1** to go to the **Manage Failed Tasks** page. In the **Status** column, hover the cursor over to view the failure cause. You can delete failed tasks by referring to **Viewing Failed MRS Tasks**.

Figure 4-1 Failed task management

Table 4-5 lists the error codes of MRS cluster creation failures.

Table 4-5 Error codes

Error Code	Description
MRS.101	Insufficient quota to meet your request. Contact customer service to increase the quota.
MRS.102	The token cannot be null or invalid. Try again later or contact customer service.
MRS.103	Invalid request. Try again later or contact customer service.
MRS.104	Insufficient resources. Try again later or contact customer service.
MRS.105	Insufficient IP addresses in the existing subnet. Try again later or contact customer service.
MRS.201	Failed due to an ECS error. Try again later or contact customer service.

Error Code	Description
MRS.202	Failed due to an IAM error. Try again later or contact customer service.
MRS.203	Failed due to a VPC error. Try again later or contact customer service.
MRS.400	MRS system error. Try again later or contact customer service.

4.4 Creating a Custom Topology Cluster

The analysis cluster, streaming cluster, and hybrid cluster provided by MRS use fixed templates to deploy cluster processes. Therefore, you cannot customize service processes on management nodes and control nodes. If you want to customize the cluster deployment, set **Cluster Type** to **Custom** when creating a cluster. In this way, you can customize the deployment mode of process instances on the management nodes and control nodes in the cluster. Only MRS 3.x and later versions support the creation of clusters in a custom topology.

A custom cluster provides the following functions:

- Separated deployment of the management and control roles: The management role and control role are deployed on different Master nodes.
- Co-deployment of the management and control roles: The management and control roles are co-deployed on the Master node.
- ZooKeeper is deployed on an independent node to improve reliability.
- Components are deployed separately to avoid resource contention.

Roles in an MRS cluster:

- Management Node (MN): is the node to install Manager (the management system of the MRS cluster). It provides a unified access entry. Manager centrally manages nodes and services deployed in the cluster.
- Control Node (CN): controls and monitors how data nodes store and receive data, and send process status, and provides other public functions. Control nodes of MRS include HMaster, HiveServer, ResourceManager, NameNode, JournalNode, and SlapdServer.
- Data Node (DN): A data node executes the instructions sent by the management node, reports task status, stores data, and provides other public functions. Data nodes of MRS include DataNode, RegionServer, and NodeManager.

Customizing a Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Custom Config** tab.

Step 4 Configure basic cluster information. For details about the parameters, see **Software Configurations**.

- **Region**: Retain the default value.
- **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing, for example, **mrs_20180321**.
- **Cluster Version**: Currently, only MRS 3.x are supported.
- **Cluster Type**: Select **Custom** and select components as required.

Step 5 Click **Next**. Configure hardware information.

- **AZ**: Retain the default value.
- **VPC**: Retain the default value. If there is no available VPC, click **View VPC** to access the VPC console and create a new VPC.
- **Subnet**: Retain the default value.
- **Security Group**: Select **Auto create**.
- **EIP**: Select **Bind later**.
- **Enterprise Project**: Retain the default value.
- **CPU Architecture**: Retain the default value. This parameter is unavailable in MRS 3.x.
- **Common Node**: For details, see **Custom Cluster Template Description**.
- **Instance Specifications**: Click  to configure the instance specifications, system disk and data disk storage types, and storage space.
- **Instance Count**: Adjust the number of cluster instances based on the service volume. For details, see **Table 4-7**.
- **Topology Adjustment**: If the deployment mode in the **Common Node** does not meet the requirements, you need to manually install some instances that are not deployed by default, or you need to manually install some instances, set **Topology Adjustment** to **Enable** and adjust the instance deployment mode based on service requirements. For details, see **Topology Adjustment for a Custom Cluster**.

Step 6 Click **Next** and set advanced options.

For details about the parameters, see **(Optional) Advanced Configuration**.

Step 7 Click **Create Now**.

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Step 8 Click **Back to Cluster List** to view the cluster status.

It takes some time to create a cluster. The initial status of the cluster is **Starting**. After the cluster has been created successfully, the cluster status becomes **Running**.

----End

Custom Cluster Template Description

Table 4-6 Common templates for custom clusters

Common Node	Description	Node Range
Compact	The management role and control role are deployed on the Master node, and data instances are deployed in the same node group. This deployment mode applies to scenarios where the number of control nodes is less than 100, reducing costs.	• The number of Master nodes is greater than or equal to 3 and less than or equal to 11. • The total number of node groups is less than or equal to 10, and the total number of nodes in non-Master node groups is less than or equal to 10,000.
OMS-separate	The management role and control role are deployed on different Master nodes, and data instances are deployed in the same node group. This deployment mode is applicable to a cluster with 100 to 500 nodes and delivers better performance in high-concurrency load scenarios.	• The number of Master nodes is greater than or equal to 5 and less than or equal to 11. • The total number of node groups is less than or equal to 10, and the total number of nodes in non-Master node groups is less than or equal to 10,000.
Full-size	The management role and control role are deployed on different Master nodes, and data instances are deployed in different node groups. This deployment mode is applicable to a cluster with more than 500 nodes. Components can be deployed separately, which can be used for a larger cluster scale.	• The number of Master nodes is greater than or equal to 9 and less than or equal to 11. • The total number of node groups is less than or equal to 10, and the total number of nodes in non-Master node groups is less than or equal to 10,000.

Table 4-7 Node deployment scheme of a customized MRS cluster

Node Deployment Principle		Applicable Scenario	Networking Rule
Management nodes, control nodes, and data nodes are deployed separately. (This scheme requires at least eight nodes.)	MN × 2 + CN × 9 + DN × n	(Recommended) This scheme is used when the number of data nodes is 500–2000.	• If the number of nodes in a cluster exceeds 200, the nodes are distributed to different subnets and the subnets are interconnected with each other in Layer 3 using core switches. Each subnet can contain a maximum of 200 nodes and the allocation of nodes to different subnets must be balanced. • If the number of nodes is less than 200, the nodes in the cluster are deployed in the same subnet and the nodes are interconnected with each other in Layer 2 using aggregation switches.
	MN × 2 + CN × 5 + DN × n	(Recommended) This scheme is used when the number of data nodes is 100–500.	
	MN × 2 + CN × 3 + DN × n	(Recommended) This scheme is used when the number of data nodes is 30–100.	
The management nodes and control nodes are deployed together, and the data nodes are deployed separately.	(MN+CN) × 3 + DN × n	(Recommended) This scheme is used when the number of data nodes is 3–30.	Nodes in the cluster are deployed in the same subnet and are interconnected with each other at Layer 2 through aggregation switches.

Node Deployment Principle	Applicable Scenario	Networking Rule
The management nodes, control nodes, and data nodes are deployed together.	• This scheme is applicable to a cluster having fewer than 6 nodes. • This scheme requires at least three nodes. **NOTE** This template is not recommended in the production environment or commercial environment. • If management, control, and data nodes are co-deployed, cluster performance and reliability are greatly affected. • If the number of nodes meet the requirements, deploy data nodes separately. • If the number of nodes is insufficient to support separately deployed data nodes, use the dual-plane networking mode for this scenario. The traffic of the management network is isolated from that of the service network to prevent excessive data volumes on the service plane, ensuring correct delivery of management operations.	Nodes in the cluster are deployed in the same subnet and are interconnected with each other at Layer 2 through aggregation switches.

Topology Adjustment for a Custom Cluster

Table 4-8 Topology adjustment

Service	Dependency	Role	Role Deployment Suggestions	Description
OMSServer	-	OMSServer	This role can be deployed it on the Master node and cannot be modified.	-

Service	Dependency	Role	Role Deployment Suggestions	Description
ClickHouse	Depends on ZooKeeper.	CHS (ClickHouseServer)	This role can be deployed on all nodes. Number of role instances to be deployed: an even number ranging from 2 to 256	A non-Master node group with this role assigned is considered as a Core node.
		CLB (ClickHouseBalancer)	This role can be deployed on all nodes. Number of role instances to be deployed: 2 to 256	-
ZooKeeper	-	QP(quorumpeer)	This role can be deployed on the Master node only. Number of role instances to be deployed: 3 to 9, with the step size of 2	-
Hadoop	Depends on ZooKeeper.	NN(NameNode)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	The NameNode and ZKFC processes are deployed on the same server for cluster HA.
		HFS (HttpFS)	This role can be deployed on the Master node only. Number of role instances to be deployed: 0 to 10	-
		JN(JournalNode)	This role can be deployed on the Master node only. Number of role instances to be deployed: 3 to 60, with the step size of 2	-

Service	Dependency	Role	Role Deployment Suggestions	Description
		DN(DataNode)	This role can be deployed on all nodes. Number of role instances to be deployed: 3 to 10,000	A non-Master node group with this role assigned is considered as a Core node.
		RM(ResourceManager)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	-
		NM(NodeManager)	This role can be deployed on all nodes. Number of role instances to be deployed: 3 to 10,000	-
		JHS(JobHistoryServer)	This role can be deployed on the Master node only. Number of role instances to be deployed: 1 to 2	-
		TLS(TimelineServer)	This role can be deployed on the Master node only. Number of role instances to be deployed: 0 to 1	-
Presto	Depends on Hive.	PCD(Coordinator)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	-

Service	Dependency	Role	Role Deployment Suggestions	Description
		PWK(Worker)	This role can be deployed on all nodes. Number of role instances to be deployed: 1 to 10,000	-
Spark2x	• Depends on Hadoop. • Depends on Hive. • Depends on ZooKeeper.	JS2X(JDBCServer2x)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2 to 10	-
		JH2X(JobHistory2x)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	-
		SR2X(SparkResource2x)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2 to 50	-
		IS2X(IndexServer2x)	(Optional) This role can be deployed on the Master node only. Number of role instances to be deployed: 0 to 2, with the step size of 2	-
HBase	Depends on Hadoop.	HM(HMaster)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	-

Service	Dependency	Role	Role Deployment Suggestions	Description
		TS(ThriftServer)	This role can be deployed on all nodes. Number of role instances to be deployed: 0 to 10,000	-
		RT(RESTServer)	This role can be deployed on all nodes. Number of role instances to be deployed: 0 to 10,000	-
		RS(RegionServer)	This role can be deployed on all nodes. Number of role instances to be deployed: 3 to 10,000	-
		TS1(Thrift1Server)	This role can be deployed on all nodes. Number of role instances to be deployed: 0 to 10,000	If the Hue service is installed in a cluster and HBase needs to be used on the Hue web UI, install this instance for the HBase service.
Hive	• Depends on Hadoop. • Depends on DBService.	MS(MetaStore)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2 to 10	-
		WH(WebHCat)	This role can be deployed on the Master node only. Number of role instances to be deployed: 1 to 10	-

Service	Dependency	Role	Role Deployment Suggestions	Description
		HS(HiveServer)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2 to 80	-
Hue	Depends on DBService.	H(Hue)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	-
Sqoop	Depends on Hadoop.	SC(Sqoop Client)	This role can be deployed on all nodes. Number of role instances to be deployed: 1 to 10,000	-
Kafka	Depends on ZooKeeper.	B(Broker)	This role can be deployed on all nodes. Number of role instances to be deployed: 3 to 10,000	-
Flume	-	MS(MonitorServer)	This role can be deployed on the Master node only. Number of role instances to be deployed: 1 to 2	-
		F(Flume)	This role can be deployed on all nodes. Number of role instances to be deployed: 1 to 10,000	A non-Master node group with this role assigned is considered as a Core node.

Service	Dependency	Role	Role Deployment Suggestions	Description
Tez	• Depends on Hadoop. • Depends on DBService. • Depends on ZooKeeper.	TUI(TezUI)	This role can be deployed on the Master node only. Number of role instances to be deployed: 1 to 2	-
Flink	• Depends on ZooKeeper. • Depends on Hadoop.	FR(FlinkResource)	This role can be deployed on all nodes. Number of role instances to be deployed: 1 to 10,000	-
		FS(FlinkServer)	This role can be deployed on all nodes. Number of role instances to be deployed: 0 to 2	-
Oozie	• Depends on Hadoop. • Depends on DBService. • Depends on ZooKeeper.	O(oozie)	This role can be deployed on the Master node only. Number of role instances to be deployed: 2	-

Service	Dependency	Role	Role Deployment Suggestions	Description
Impala	• Depends on Hadoop. • Depends on Hive. • Depends on DBService. • Depends on ZooKeeper.	StateStore	This role can be deployed on the Master node only. Number of role instances to be deployed: 1	-
		Catalog	This role can be deployed on the Master node only. Number of role instances to be deployed: 1	-
		Impalad	This role can be deployed on all nodes. Number of role instances to be deployed: 1 to 10,000	-
Kudu	-	KuduMaster	This role can be deployed on the Master node only. Number of role instances to be deployed: 3 or 5	-
		KuduTserver	This role can be deployed on all nodes. Number of role instances to be deployed: 3 to 10,000	-
Ranger	Depends on DBService.	RA(RangerAdmin)	This role can be deployed on the Master node only. Number of role instances to be deployed: 1 to 2	-

Service	Dependency	Role	Role Deployment Suggestions	Description
		USC(User Sync)	This role can be deployed on the Master node only. Number of role instances to be deployed: 1	-
		TSC (TagSync)	This role can be deployed on all nodes. Number of role instances to be deployed: 0 to 1	-

4.5 Adding a Tag to a Cluster

Tags are used to identify clusters. Adding tags to clusters can help you identify and manage your cluster resources.

You can add a maximum of 10 tags to a cluster when creating the cluster or add them on the details page of the created cluster.

A tag consists of a tag key and a tag value. **Table 4-9** provides tag key and value requirements.

Table 4-9 Tag key and value requirements

Parameter	Requirement	Example
Key	A tag key cannot be left blank. A tag key must be unique in a cluster. A tag key contains a maximum of 36 characters. A tag value cannot contain special characters (=*<>\,|/) or start or end with spaces.	Organization

Parameter	Requirement	Example
Value	A tag value contains a maximum of 43 characters. A tag value cannot contain special characters (=*<>\,|/) or start or end with spaces. This parameter can be left blank.	Apache

Adding Tags to a Cluster

You can perform the following operations to add tags to a cluster when creating the cluster.

1. Log in to the MRS console.
2. Click **Create Cluster**. The corresponding page is displayed.
3. Click the **Custom Config** tab.
4. Configure the cluster software and hardware by referring to **Creating a Custom Cluster**.
5. On the **Set Advanced Options** tab page, add a tag.

   Enter the key and value of a tag to be added.

   You can add a maximum of 10 tags to a cluster and use intersections of tags to search for the target cluster.

   NOTE

   You can also add tags to existing clusters. For details, see **Managing Tags**.

Searching for the Target Cluster

On the **Active Clusters** page, search for the target cluster by tag key or tag value.

1. Log in to the MRS console.
2. In the upper right corner of the **Active Clusters** page, click **Search by Tag** to access the search page.
3. Enter the tag of the cluster to be searched.

   You can select a tag key or tag value from their drop-down lists. When the tag key or tag value is exactly matched, the system can automatically locate the target cluster. If you enter multiple tags, their intersections are used to search for the cluster.
4. Click **Search**.

   The system searches for the target cluster by tag key or value.

Managing Tags

You can view, add, modify, and delete tags on the **Tags** tab page of the cluster.

1. Log in to the MRS console.
2. On the **Active Clusters** page, click the name of a cluster for which you want to manage tags.

   The cluster details page is displayed.
3. Click the **Tags** tab and view, add, modify, and delete tags on the tab page.
   - View

     On the **Tags** tab page, you can view details about tags of the cluster, including the number of tags and the key and value of each tag.
   - Add

     Click **Add Tag** in the upper left corner. In the displayed **Add Tag** dialog box, enter the key and value of the tag to be added, and click **OK**.
   - Modify

     In the **Operation** column of the tag, click **Edit**. In the displayed **Edit Tag** page, enter new tag key and value and click **OK**.
   - Delete

     In the **Operation** column of the tag, click **Delete**. After confirmation, click **OK** in the displayed page for deleting a tag.

     NOTE

     MRS cluster tag updates will be synchronized to every ECS in the cluster. You are advised not to modify ECS tags on the ECS console to prevent inconsistency between ECS tags and MRS cluster tags. If the number of tags of an ECS in the MRS cluster reaches the upper limit, you cannot create any tag for the MRS cluster.

4.6 Communication Security Authorization

MRS clusters provision, manage, and use big data components through the management console. Big data components are deployed in a user's VPC. If the MRS management console needs to directly access big data components deployed in the user's VPC, you need to enable the corresponding security group rules after you have obtained user authorization. This authorization process is called secure communications.

If the secure communications function is not enabled, MRS clusters cannot be created. If you disable the communication after a cluster is created, the cluster status will be **Network channel is not authorized** and the following functions will be affected:

- Functions, such as big data component installation, cluster scale-out/scale-in, and Master node specification upgrade, are unavailable.
- The cluster running status, alarms, and events cannot be monitored.
- The node management, component management, alarm management, file management, job management, patch management, and tenant management functions on the cluster details page are unavailable.
- The Manager page and the website of each component cannot be accessed.

After the secure communications function is enabled again, the cluster status is restored to **Running**, and the preceding functions become available. For details, see **Enabling Secure Communications for Clusters with This Function Disabled**.

If the security group rules authorized in the cluster are insufficient for you to provision, manage, and use big data components, is displayed on the right of **Secure Communications**. In this case, click **Update** to update the security group rules. For details, see **Update**.

Enabling Secure Communications During Cluster Creation

Step 1 Log in to the MRS console.

Step 2 Click Create **Cluster**. The corresponding page is displayed.

Step 3 Click **Quick Config** or **Custom Config**.

Step 4 Configure cluster information by referring to **Creating a Custom Cluster**.

Step 5 In the **Secure Communications** area of the **Advanced Settings** tab page, select **Enable**.

Step 6 Click **Create Now**.

If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

----End

Disabling Secure Communications After a Cluster Is Created

Step 1 Log in to the MRS console.

Step 2 In the active cluster list, click the name of the cluster for which you want to disable secure communications.

The cluster details page is displayed.

Step 3 Click the switch on the right of **Secure Communications** to disable authorization. In the dialog box that is displayed, click **OK**.

After the authorization is disabled, the cluster status changes to **Network channel unauthorized**, and some functions of the cluster are unavailable. Exercise caution when performing this operation.

----End

Enabling Secure Communications for Clusters with This Function Disabled

Step 1 Log in to the MRS console.

Step 2 In the active cluster list, click the name of the cluster for which you want to enable secure communications.

The cluster details page is displayed.

Step 3 Click the switch on the right of **Secure Communications** to enable the function.

After the function is enabled, the cluster status changes to **Running**.

----End

Update

If the security group rules authorized in the cluster are insufficient for you to provision, manage, and use big data components, is displayed on the right of **Secure Communications**. In this case, click **Update** to update the security group rules. For details, see **Update**.

Step 1 Log in to the MRS console.

Step 2 In the active cluster list, click the name of the cluster for which you want to update secure communications.

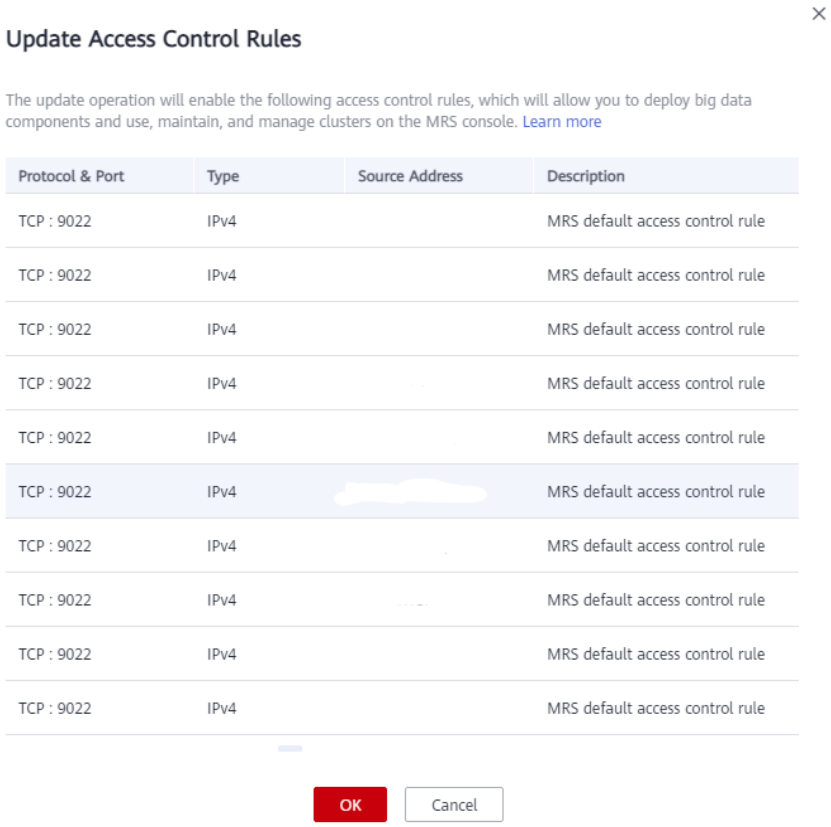
The cluster details page is displayed.

Step 3 Click **Update** on the right of **Secure Communications**.

Figure 4-2 Update

Step 4 Click **OK**.

Figure 4-3 Updating access control rules



----End

4.7 Configuring an Auto Scaling Rule

Background

In big data application scenarios, especially real-time data analysis and processing, the number of cluster nodes needs to be dynamically adjusted according to data volume changes to provide the required number of resources. The auto scaling function of MRS enables the task nodes of a cluster to be automatically scaled to match cluster loads. If the data volume changes periodically, you can configure an auto scaling rule so that the number of task nodes can be automatically adjusted in a fixed period of time before the data volume changes.

- Auto scaling rules: You can increase or decrease task nodes based on real-time cluster loads. Auto scaling will be triggered with a certain delay when the data volume changes.
- Resource plans: Set the task node quantity based on the time range. If the data volume changes periodically, you can create resource plans to resize the cluster before the data volume changes, thereby avoiding delays in increasing or decreasing resources.

You can configure either auto scaling rules or resource plans or both to trigger auto scaling. Configuring both resource plans and auto scaling rules improves the cluster node scalability to cope with occasionally unexpected data volume peaks.

In some service scenarios, resources need to be reallocated or service logic needs to be modified after cluster scale-out or scale-in. If you manually scale out or scale in a cluster, you can log in to cluster nodes to reallocate resources or modify service logic. If you use auto scaling, MRS enables you to customize automation scripts for resource reallocation and service logic modification. Automation scripts can be executed before and after auto scaling and automatically adapt to service load changes, all of which eliminates manual operations. In addition, automation scripts can be fully customized and executed at various moments, meeting your personalized requirements and improving auto scaling flexibility.

- Auto scaling rules:
  - You can set a maximum of five rules for scaling out or in a cluster, respectively.
  - The system determines the scale-out and then scale-in based on your configuration sequence. Important policies take precedence over other policies to prevent repeated triggering when the expected effect cannot be achieved after a scale-out or scale-in.
  - Comparison factors include greater than, greater than or equal to, less than, and less than or equal to.
  - Cluster scale-out or scale-in can be triggered only after the configured metric threshold is reached for consecutive 5*n* (the default value of *n* is 1) minutes.
  - After each scale-out or scale-in, there is a cooling duration is greater than 0, and lasts 20 minutes by defaults.
  - In each cluster scale-out or scale-in, at least one node and at most 100 nodes can be added or reduced.

- Resource plans (setting the number of Task nodes by time range):
  - You can specify a Task node range (minimum number to maximum number) in a time range. If the number of Task nodes is beyond the Task node range in a resource plan, the system triggers cluster scale-out or scale-in.
  - You can set a maximum of five resource plans for a cluster.
  - A resource plan cycle is by day. The start time and end time can be set to any time point between 00:00 and 23:59. The start time must be at least 30 minutes earlier than the end time. Time ranges configured for different resource plans cannot overlap.
  - After a resource plan triggers cluster scale-out or scale-in, there is 10-minute cooling duration. Auto scaling will not be triggered again within the cooling time.
  - When a resource plan is enabled, the number of Task nodes in the cluster is limited to the default node range configured by you in other time periods except the time period configured in the resource plan.
  - If the resource plan is not enabled, the number of Task nodes is not limited to the default node range.
- Automation scripts:
  - You can set an automation script so that it can automatically run on cluster nodes when auto scaling is triggered.
  - You can set a maximum number of 10 automation scripts for a cluster.
  - You can specify an automation script to be executed on one or more types of nodes.
  - Automation scripts can be executed before or after scale-out or scale-in.
  - Before using automation scripts, upload them to a cluster VM or OBS file system in the same region as the cluster. The automation scripts uploaded to the cluster VM can be executed only on the existing nodes. If you want to make the automation scripts run on the new nodes, upload them to the OBS file system.

Accessing the Auto Scaling Configuration Page

You can configure an auto scaling rule on the **Set Advanced Options** page during cluster creation or on the **Nodes** page after the cluster is created.

Configuring an auto scaling rule when creating a cluster

Step 1 Log in to the MRS console.

Step 2 When you a cluster containing task nodes, configure the cluster software and hardware information by referring to **Creating a Custom Cluster**. Then, on the **Set Advanced Options** page, enable **Analysis Task** and configure or modify auto scaling rules and resource plans.

You can configure the auto scaling rules by referring to the following scenarios:

- **Scenario 1: Using Auto Scaling Rules Alone**
- **Scenario 2: Using Resource Plans Alone**

- **Scenario 3: Using Both Auto Scaling Rules and Resource Plans**

----End

Configure an auto scaling rule for an existing cluster

Step 1 Log in to the MRS console.

Step 2 In the navigation pane on the left, choose **Clusters** > **Active Clusters** and click the name of a running cluster to go to the cluster details page.

Step 3 Click the **Nodes** tab and then **Auto Scaling** in the **Operation** column of the task node group. The **Auto Scaling** page is displayed.

NOTE

- If no task node exists in the cluster, click **Configure Task Node** to add one and then configure the auto scaling rules.
- For MRS 3.*x* or later, **Configure Task Node** is available only for analysis clusters, streaming clusters, and hybrid clusters. For details about how to add a task node for a custom cluster of MRS 3.*x* or later, see **Adding a Task Node**.

Step 4 Enable **Auto Scaling** and configure or modify auto scaling rules and resource plans.

You can configure the auto scaling rules by referring to the following scenarios:

- **Scenario 1: Using Auto Scaling Rules Alone**
- **Scenario 2: Using Resource Plans Alone**
- **Scenario 3: Using Both Auto Scaling Rules and Resource Plans**

----End

Scenario 1: Using Auto Scaling Rules Alone

The following is an example scenario:

The number of nodes needs to be dynamically adjusted based on the Yarn resource usage. When the memory available for Yarn is less than 20% of the total memory, five nodes need to be added. When the memory available for Yarn is greater than 70% of the total memory, five nodes need to be removed. The number of nodes in a task node group ranges from 1 to 10.

Step 1 Go to the **Auto Scaling** page to configure auto scaling rules.

- Configure the **Default Range** parameter.

  Enter a task node range, in which auto scaling is performed. This constraint applies to all scale-in and scale-out rules. The maximum value range allowed is 0 to 500.

  The value range in this example is 1 to 10.

- Configure an auto scaling rule.

  To enable **Auto Scaling**, you must configure a scale-out or scale-in rule.

  a. Select **Scale-Out** or **Scale-In**.

  b. Click **Add Rule**.

  c. Configure the **Rule Name**, **If**, **Last for**, **Add**, and **Cooldown Period** parameters.

d. Click **OK**.

You can view, edit, or delete the rules you configured in the **Scale-out** or **Scale-in** area on the **Auto Scaling** page. You can click **Add Rule** to configure multiple rules.

Step 2 (Optional) Configure automation scripts.

Set **Advanced Settings** to **Configure** and click **Created**, or click **Add Automation Script** to go to the **Automation Script** page.

MRS 3.*x* does not support this operation.

1. Set the following parameters: **Name**, **Script Path**, **Execution Node**, **Parameter**, **Executed**, and **Action upon Failure**. For details about the parameters, see **Table 4-12**.
2. Click **OK** to save the automation script configurations.

Step 3 Click **OK**.

NOTE

If you want to configure an auto scaling rule for an existing cluster, select **I agree to authorize MRS to scale out or in nodes based on the above rule**.

----End

Scenario 2: Using Resource Plans Alone

If the data volume changes regularly every day and you want to scale out or in a cluster before the data volume changes, you can create resource plans to adjust the number of Task nodes as planned in the specified time range.

Example:

A real-time processing service sees a sharp increase in data volume from 7:00 to 13:00 every day. Assume that an MRS streaming cluster is used to process the service data. Five task nodes are required from 7:00 to 13:00, while only two are required at other time.

Step 1 Go to the **Auto Scaling** page to configure a resource plan.

1. For example, the **Default Range** is set to **2-2**, indicating that the number of Task nodes is fixed to 2 except the time range specified in the resource plan.
2. Click **Configure Node Range for Specific Time Range** under **Default Range** or **Add Resource Plan**.
3. Configure **Time Range** and **Node Range**.

   For example, set **Time Range** to **07:00-13:00**, and **Node Range** to **5-5**. This indicates that the number of task nodes is fixed at 5 from 07:00 to 13:00.

   For details about parameter configurations, see **Table 4-11**. You can click **Configure Node Range for Specific Time Range** to configure multiple resource plans.

NOTE

- If you do not set **Node Range**, its default value will be used.
- If you set both **Node Range** and **Time Range**, the node range you set will be used during the time range you set, and the default node range will be used beyond the time range you set. If the time is not within the configured time range, the default range is used.

Step 2 (Optional) Configure automation scripts.

Set **Advanced Settings** to **Configure** and click **Created**, or click **Add Automation Script** to go to the **Automation Script** page.

MRS 3.*x* does not support this operation.

1. Set the following parameters: **Name**, **Script Path**, **Execution Node**, **Parameter**, **Executed**, and **Action upon Failure**. For details about the parameters, see **Table 4-12**.
2. Click **OK** to save the automation script configurations.

Step 3 Click **OK**.

NOTE

If you want to configure an auto scaling rule for an existing cluster, select **I agree to authorize MRS to scale out or in nodes based on the above rule**.

----End

Scenario 3: Using Both Auto Scaling Rules and Resource Plans

If the data volume is not stable and the expected fluctuation may occur, the fixed Task node range cannot guarantee that the requirements in some service scenarios are met. In this case, it is necessary to adjust the number of Task nodes based on the real-time loads and resource plans.

The following is an example scenario:

A real-time processing service sees an unstable increase in data volume from 7:00 to 13:00 every day. For example, 5 to 8 task nodes are required from 7:00 to 13:00, and 2 to 4 are required beyond this period. Therefore, you can set an auto scaling rule based on a resource plan. When the data volume exceeds the expected value, the number of Task nodes can be adjusted if resource loads change, without exceeding the node range specified in the resource plan. When a resource plan is triggered, the number of nodes is adjusted within the specified node range with minimum affect. That is, increase nodes to the upper limit and decrease nodes to the lower limit.

Step 1 Go to the **Auto Scaling** page to configure auto scaling rules.

- **Default Range**

  Enter a task node range, in which auto scaling is performed. This constraint applies to all scale-in and scale-out rules.

  For example, this parameter is set to **2-4** in this scenario.

- **Auto Scaling**

  To enable **Auto Scaling**, you must configure a scale-out or scale-in rule.

a. Select **Scale-Out** or **Scale-In**.

b. Click **Add Rule**. The **Add Rule** page is displayed.

c. Configure the **Rule Name**, **If**, **Last for**, **Add**, and **Cooldown Period** parameters.

d. Click **OK**.

You can view, edit, or delete the rules you configured in the **Scale-out** or **Scale-in** area on the **Auto Scaling** page.

Step 2 Configure a resource plan.

1. Click **Configure Node Range for Specific Time Range** under **Default Range** or **Add Resource Plan**.
2. Configure **Time Range** and **Node Range**.

   For example, **Time Range** is set to **07:00-13:00** and **Node Range** to **5-8**.

   For details about parameter configurations, see **Table 4-11**. You can click **Configure Node Range for Specific Time Range** or **Add Resource Plan** to configure multiple resource plans.

   NOTE

   - If you do not set **Node Range**, its default value will be used.
   - If you set both **Node Range** and **Time Rang**e, the node range you set will be used during the time range you set, and the default node range will be used beyond the time range you set. If the time is not within the configured time range, the default range is used.

Step 3 (Optional) Configure automation scripts.

Set **Advanced Settings** to **Configure** and click **Created**, or click **Add Automation Script** to go to the **Automation Script** page.

MRS 3.*x* does not support this operation.

1. Set the following parameters: **Name**, **Script Path**, **Execution Node**, **Parameter**, **Executed**, and **Action upon Failure**. For details about the parameters, see **Table 4-12**.
2. Click **OK** to save the automation script configurations.

Step 4 Click **OK**.

NOTE

If you want to configure an auto scaling rule for an existing cluster, select **I agree to authorize MRS to scale out or in nodes based on the above rule**.

----End

Related Information

When adding a rule, you can refer to **Table 4-10** to configure the corresponding metrics.

Table 4-10 Auto scaling metrics

Cluster Type	Metric	Value Type	Description
Streaming cluster	StormSlotAvailable	Integer	Number of available Storm slots Value range: 0 to 2147483646
	StormSlotAvailablePercentage	Percentage	Percentage of available Storm slots, that is, the proportion of the available slots to total slots Value range: 0 to 100
	StormSlotUsed	Integer	Number of the used Storm slots Value range: 0 to 2147483646
	StormSlotUsedPercentage	Percentage	Percentage of the used Storm slots, that is, the proportion of the used slots to total slots Value range: 0 to 100
	StormSupervisorMemAverageUsage	Integer	Average memory usage of the Supervisor process of Storm Value range: 0 to 2147483646
	StormSupervisorMemAverageUsagePercentage	Percentage	Average percentage of the used memory of the Supervisor process of Storm to the total memory of the system Value range: 0 to 100
	StormSupervisorCPUAverageUsagePercentage	Percentage	Average percentage of the used CPUs of the Supervisor process of Storm to the total CPUs Value range: 0 to 6000
Analysis cluster	YARNAppPending	Integer	Number of pending tasks on YARN Value range: 0 to 2147483646
	YARNAppPendingRatio	Ratio	Ratio of pending tasks on Yarn, that is, the ratio of pending tasks to running tasks on Yarn Value range: 0 to 2147483646
	YARNAppRunning	Integer	Number of running tasks on Yarn Value range: 0 to 2147483646
	YARNContainerAllocated	Integer	Number of containers allocated to Yarn Value range: 0 to 2147483646

Cluster Type	Metric	Value Type	Description
	YARNContainerPending	Integer	Number of pending containers on Yarn Value range: 0 to 2147483646
	YARNContainerPendingRatio	Ratio	Ratio of pending containers on Yarn, that is, the ratio of pending containers to running containers on Yarn. Value range: 0 to 2147483646
	YARNCPUAllocated	Integer	Number of virtual CPUs (vCPUs) allocated to Yarn Value range: 0 to 2147483646
	YARNCPUAvailable	Integer	Number of available vCPUs on Yarn Value range: 0 to 2147483646
	YARNCPUAvailablePercentage	Percentage	Percentage of available vCPUs on Yarn, that is, the proportion of available vCPUs to total vCPUs Value range: 0 to 100
	YARNCPUPending	Integer	Number of pending vCPUs on Yarn Value range: 0 to 2147483646
	YARNMemoryAllocated	Integer	Memory allocated to Yarn. The unit is MB. Value range: 0 to 2147483646
	YARNMemoryAvailable	Integer	Available memory on Yarn. The unit is MB. Value range: 0 to 2147483646
	YARNMemoryAvailablePercentage	Percentage	Percentage of available memory on Yarn, that is, the proportion of available memory to total memory on Yarn Value range: 0 to 100
	YARNMemoryPending	Integer	Pending memory on Yarn Value range: 0 to 2147483646

NOTE

- When the value type is percentage or ratio in Table 4-10, the valid value can be accurate to percentile. The percentage metric value is a decimal value with a percent sign (%) removed. For example, 16.80 represents 16.80%.
- Hybrid clusters support all metrics of analysis and streaming clusters.

When adding a resource plan, you can set parameters by referring to Table 4-11.

Table 4-11 Configuration items of a resource plan

Configuration Item	Description
Time Range	Start time and End time of a resource plan are accurate to minutes, with the value ranging from **00:00** to **23:59**. For example, if a resource plan starts at 8:00 and ends at 10:00, set this parameter to 8:00-10:00. The end time must be at least 30 minutes later than the start time.
Node Range	The number of nodes in a resource plan ranges from **0** to **500**. In the time range specified in the resource plan, if the number of Task nodes is less than the specified minimum number of nodes, it will be increased to the specified minimum value of the node range at a time. If the number of Task nodes is greater than the maximum number of nodes specified in the resource plan, the auto scaling function reduces the number of Task nodes to the maximum value of the node range at a time. The minimum number of nodes must be less than or equal to the maximum number of nodes.

NOTE

- When a resource plan is enabled, the **Default Range** value on the auto scaling page forcibly takes effect beyond the time range specified in the resource plan. For example, if **Default Range** is set to **1-2**, **Time Range** is between **08:00-10:00**, and **Node Range** is **4-5** in a resource plan, the number of Task nodes in other periods (0:00-8:00 and 10:00-23:59) of a day is forcibly limited to the default node range (1 to 2). If the number of nodes is greater than 2, auto scale-in is triggered; if the number of nodes is less than 1, auto scale-out is triggered.
- When a resource plan is not enabled, the **Default Range** takes effect in all time ranges. If the number of nodes is not within the default node range, the number of Task nodes is automatically increased or decreased to the default node range.
- Time ranges of resource plans cannot be overlapped. The overlapped time range indicates that two effective resource plans exist at a time point. For example, if resource plan 1 takes effect from **08:00** to **10:00** and resource plan 2 takes effect from **09:00** to **11:00**, the time range between **09:00** to **10:00** is overlapped.
- The time range of a resource plan must be on the same day. For example, if you want to configure a resource plan from **23:00** to **01:00** in the next day, configure two resource plans whose time ranges are **23:00-00:00** and **00:00-01:00**, respectively.

When adding an automation script, you can set related parameters by referring to Table 4-12.

Table 4-12 Configuration items of an automation script

Configuration Item	Description
Name	Automation script name. The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and must not start with a space. The value can contain 1 to 64 characters. **NOTE** A name must be unique in the same cluster. You can set the same name for different clusters.
Script Path	Script path. The value can be an OBS file system path or a local VM path. ● An OBS file system path must start with **s3a://** and end with **.sh**, for example, **s3a://mrs-samples/*xxx*.sh**. ● A local VM path must start with a slash (/) and end with **.sh**. For example, the path of the example script for installing the Zepelin is **/opt/bootstrap/zepelin/zepelin_install.sh**.
Execution Node	Select a type of the node where an automation script is executed. **NOTE** ● If you select **Master** nodes, you can choose whether to run the script only on the active Master nodes by enabling or disabling the **Active Master** switch. ● If you enable it, the script runs only on the active Master nodes. If you disable it, the script runs on all Master nodes. This switch is disabled by default.
Parameter	Automation script parameter. The following predefined variables can be imported to obtain auto scaling information: ● **${mrs_scale_node_num}**: Number of auto scaling nodes. The value is always positive. ● **${mrs_scale_type}**: Scale-out/in type. The value can be **scale_out** or **scale_in**. ● **${mrs_scale_node_hostnames}**: Host names of the auto scaling nodes. Use commas (,) to separate multiple host names. ● **${mrs_scale_node_ips}**: IP address of the auto scaling nodes. Use commas (,) to separate multiple IP addresses. ● **${mrs_scale_rule_name}**: Name of the triggered auto scaling rule. For a resource plan, this parameter is set to **resource_plan**.

Configuration Item	Description
Executed	Time for executing an automation script. The following four options are supported: **Before scale-out**, **After scale-out**, **Before scale-in**, and **After scale-in**. **NOTE** Assume that the execution nodes include Task nodes. • The automation script executed before scale-out cannot run on the Task nodes to be added. • The automation script executed after scale-out can run on the added Task nodes. • The automation script executed before scale-in can run on Task nodes to be deleted. • The automation script executed after scale-in cannot run on the deleted Task nodes.
Action upon Failure	Whether to continue to execute subsequent scripts and scale-out/in after the script fails to be executed. **NOTE** • You are advised to set this parameter to **Continue** in the commissioning phase so that the cluster can continue the scale-out/in operation no matter whether the script is executed successfully. • If the script fails to be executed, view the log in **/var/log/Bootstrap** on the cluster VM. • The scale-in operation cannot be rolled back. Therefore, the **Action upon Failure** can only be set to **Continue** after scale-in.

NOTE

The automation script is triggered only during auto scaling. It is not triggered when the cluster node is manually scaled out or in.

4.8 Managing Data Connections

4.8.1 Configuring Data Connections

MRS data connections are used to manage external source connections used by components in a cluster. For example, if Hive metadata uses an external relational database, a data connection can be used to associate the external relational database with the Hive component.

- **Local**: Metadata is stored in the local GaussDB of a cluster. When the cluster is deleted, the metadata is also deleted. To retain the metadata, manually back up the metadata in the database in advance.
- **Data Connection**: Metadata is stored in the associated PostgreSQL or MySQL database of the RDS service in the same VPC and subnet as the current cluster. When the cluster is terminated, the metadata is not deleted. Multiple MRS clusters can share the metadata.

> NOTE
>
> When Hive metadata is switched between different clusters, MRS synchronizes only the permissions in the metadata database of the Hive component. The permission model on MRS is maintained on MRS Manager. Therefore, when Hive metadata is switched between clusters, the permissions of users or user groups cannot be automatically synchronized to MRS Manager of another cluster.

Performing Operations Before Data Connection

Step 1 Log in to the RDS console.

Step 2 Click the **Instance Management** tab and click the name of the RDS DB instance used by the MRS data connection.

Step 3 Click **Log In** in the upper right corner to log in to the instance as user **root**.

Step 4 On the home page of the instance, click **Create Database** to create a database.

Step 5 On the top of the page, choose **Account Management** > **User Management**.

> NOTE
>
> If the selected data connection is **RDS MySQL database**, ensure that the database user is user **root**. If the user is not **root**, perform **Step 5** to **Step 7**.

Step 6 Click **Create User** to create a non-root user.

Step 7 On the top of the page, choose **SQL Operations** > **SQL Query**, switch to the target database by database name, and run the following SQL statements to grant permissions to the database user. In the following statements, *${db_name}* and *${db_user}* indicate the name of the database to be connected to MRS and the name of the new user, respectively.

```
grant SELECT, INSERT on mysql.* to '${db_user}'@'%' with grant option;
grant all privileges on ${db_name}.* to '${db_user}'@'%' with grant option;
grant reload on *.* to '${db_user}'@'%' with grant option;
flush privileges;
```

Step 8 Create a data connection by referring to **Creating a Data Connection**.

----End

Creating a Data Connection

Step 1 Log in to the MRS management console, and choose **Data Connections** in the left navigation pane.

Step 2 Click **Create Data Connection**.

Step 3 Set parameters according to **Table 4-13**.

Table 4-13 Data connection parameters

Parameter	Description
Type	Type of an external source connection. • RDS for PostgreSQL database. Clusters of that support Hive can connect to this type of database. • RDS for MySQL database. Clusters of that supports Hive or Ranger can connect to this type of database.
Name	Name of a data connection.
RDS Instance	RDS database instance. This instance must be created in RDS before being referenced here, and the database must have been created. For details, see **Performing Operations Before Data Connection**. Click **View RDS Instance** to view the created instances. **NOTE** • To ensure network communications between the cluster and the PostgreSQL database, you are advised to create the instance in the same VPC and subnet as the cluster. • The inbound rule of the security group of the RDS instance must allow access of the instance to port 3306. To configure that, click the instance name on the RDS console to go to the instance management page. In **Connection Information** area, click the name of **Security Group**. On the page that is displayed, click the **Inbound Rules** tab, and click **Add Rule**. On the displayed dialog box, in **Protocol & Port** area, select **TCP** and enter port number **3306**. In **Source** area, enter the IP address of all nodes where the MetaStore instance of Hive resides. • Currently, MRS supports **PostgreSQL9.5/PostgreSQL9.6** on RDS. • Currently, MRS supports only **MySQL 5.7.*x*** on RDS.
Database	Name of the database to be connected to.
Username	Username for logging in to the database to be connected.
Password	Password for logging in to the database to be connected.

NOTE

If the selected data connection is an **RDS MySQL** database, ensure that the database user is a **root** user. If the user is not **root**, perform operations by referring to **Performing Operations Before Data Connection**.

Step 4 Click **OK**.

----End

Editing a Data Connection

Step 1 Log in to the MRS management console, and choose **Data Connections** in the left navigation pane.

Step 2 In the **Operation** column of the data connection list, click **Edit** in the row where the data connection to be edited is located.

Step 3 Modify parameters according to **Table 4-13**.

If the selected data connection has been associated with a cluster, the configuration changes will be synchronized to the cluster.

----End

Deleting a Data Connection

Step 1 Log in to the MRS management console, and choose **Data Connections** in the left navigation pane.

Step 2 In the **Operation** column of the data connection list, click **Delete** in the row where the data connection to be deleted is located.

If the selected data connection has been associated with a cluster, the deletion does not affect the cluster.

----End

Configuring a data connection during cluster creation

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The **Create Cluster** page is displayed.

Step 3 Click the **Custom Config** tab.

Step 4 In the software configuration area, set **Metadata** by referring to **Table 4-14**. For other parameters, see **Creating a Custom Cluster** for configuration and cluster creation.

Table 4-14 Data connection parameters

Parameter	Description
Metadata	Whether to use external data sources to store metadata. • **Local**: Metadata is stored in the local cluster. • **Data connections**: Metadata of external data sources is used. If the cluster is abnormal or deleted, metadata is not affected. This mode applies to scenarios where storage and compute are decoupled. Clusters that support the Hive or Ranger component support this function.
Component	This parameter is valid only when **Use External Data Sources to Store Metadata** is enabled. It indicates the type of an external data source. • Hive • Ranger
Data Connection Type	This parameter is valid only when **Use External Data Sources to Store Metadata** is enabled. It indicates the type of an external data source. • Hive supports the following data connection types: – RDS for PostgreSQL (supported for clusters of MRS 1.9.*x*) – RDS MySQL database – Local database • Ranger supports the following data connection types: – RDS MySQL database – Local database
Data Connection Instance	This parameter is valid only when **Data Connection Type** is set to **RDS PostgreSQL database** or **RDS MySQL database**. This parameter indicates the name of the connection between the MRS cluster and the RDS database. This instance must be created before being referenced here. You can click **Create Data Connection** to create a data connection. For details, see **Performing Operations Before Data Connection** and **Creating a Data Connection**.

----End

4.8.2 Configuring Ranger Data Connections

Switch the Ranger metadata of the existing cluster to the metadata stored in the RDS database. This operation enables multiple MRS clusters to share the same

metadata, and the metadata will not be deleted when the clusters are deleted. In this way, Ranger metadata migration is not required during cluster migration.

Prerequisites

You have created an RDS MySQL database instance. For details, see **Creating a Data Connection**.

NOTE

- For versions earlier than MRS 3.x, if the selected data connection is an **RDS MySQL database**, ensure that the database user is a **root** user. If the user is not **root**, create a user and grant permissions to the user by referring to **Performing Operations Before Data Connection**.
- In MRS 3.x or later, if the selected data connection is **RDS MySQL database**, the database user cannot be user **root**. In this case, create a user and grant permissions to the user by following the instructions provided in **Performing Operations Before Data Connection**.

Preparing for MySQL Database Ranger Metadata Configuration

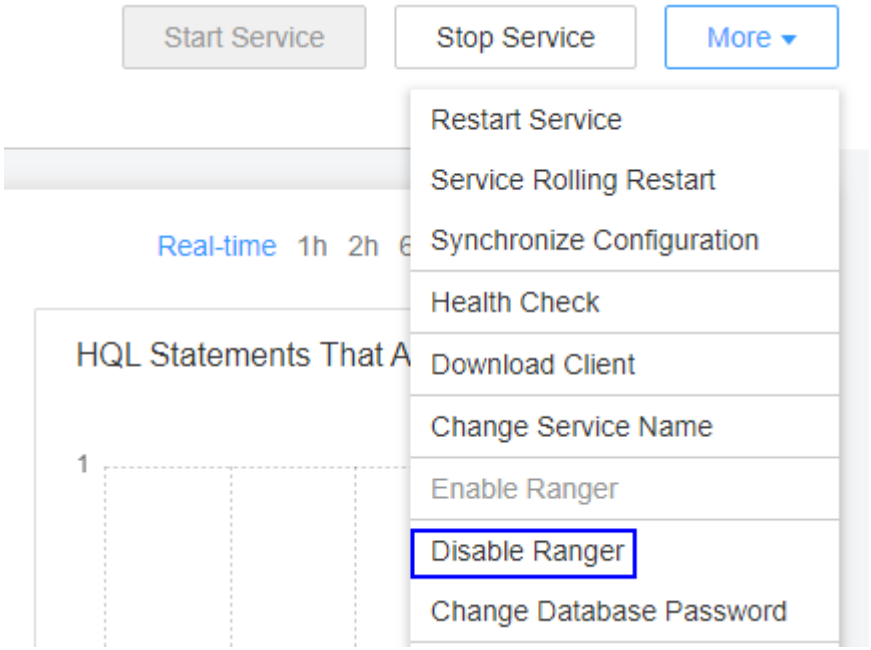
This operation is required only for **MRS 3.1.0 or later**.

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Clusters** > **Services** > *Service name*.

Currently, the following components in an MRS 3.1.*x* cluster support Ranger authentication: HDFS, HBase, Hive, Spark, Impala, Storm, and Kafka.

Step 2 In the upper right corner of the **Dashboard** page, click **More** and select **Disable Ranger**. If **Disable Ranger** is dimmed, Ranger authentication is disabled, as shown in **Figure 4-4**.

Figure 4-4 Disabling Ranger authentication

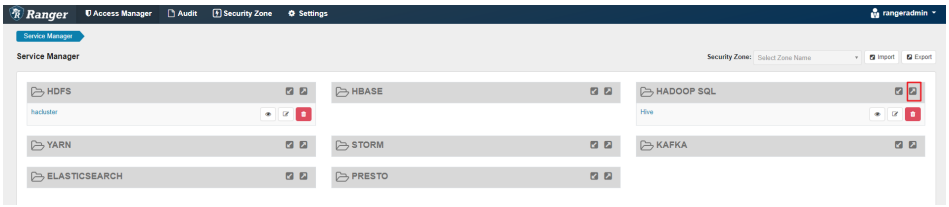


Step 3 (Optional) To use an existing authentication policy, perform this step to export the authentication policy on the Ranger web page. After the Ranger metadata is switched, you can import the existing authentication policy again. The following uses Hive as an example. After the export, a policy file in JSON format is generated in a local directory.

1. Log in to FusionInsight Manager.
2. Choose **Cluster** > **Services** > **Ranger** to go to the Ranger service overview page.
3. Click **RangerAdmin** in the **Basic Information** area to go to the Ranger web UI.

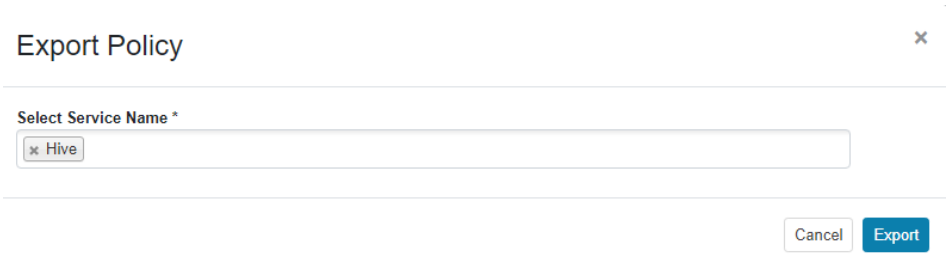
   The **admin** user in Ranger belongs to the **User** type. To view all management pages, click the username in the upper right corner and select **Log Out** to log out of the system.
4. Log in to the system as user **rangeradmin** (default password: **Rangeradmin@123**) or another user who has the Ranger administrator permissions.
5. Click the export button  in the row where the Hive component is located to export the authentication policy.

   Figure 4-5 Exporting authentication policies

6. Click **Export**. After the export is complete, a policy file in JSON format is generated in a local directory.

   Figure 4-6 Exporting Hive authentication policies

----End

Configuring a Data Connection for an MRS Cluster

Step 1 Log in to the MRS console.

Step 2 Click the name of the cluster to view its details.

Step 3 Click **Manage** on the right of **Data Connection** to go to the data connection configuration page.

Step 4 Click **Configure Data Connection** and set related parameters.

- **Component Name**: Ranger
- **Module Type**: Ranger metadata
- **Connection Type**: RDS MySQL database
- **Connection Instance**: Select a created RDS MySQL DB instance. To create a new data connection, see **Creating a Data Connection**.

Step 5 Select **I understand the consequences of performing the scale-in operation** and click **Test**.

Step 6 After the test is successful, click **OK** to complete the data connection configuration.

Step 7 Log in to FusionInsight Manager.

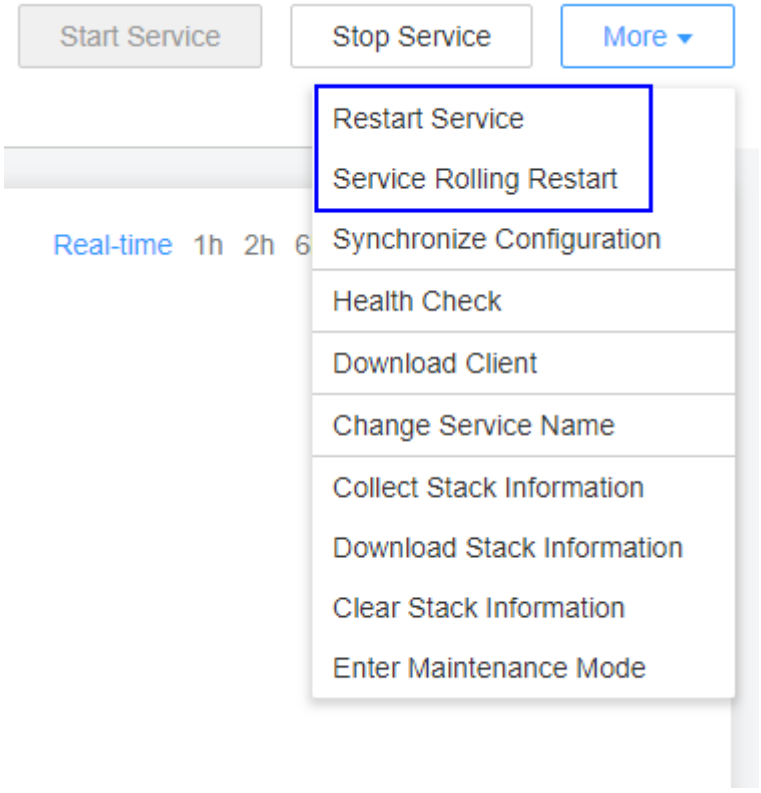
Step 8 Choose **Cluster** > **Services** > **Ranger** to go to the Ranger service overview page.

Step 9 Choose **More** > **Restart Service** or **More** > **Service Rolling Restart**.

If you choose **Restart Service**, services will be interrupted during the restart. If you select **Service Rolling Restart**, rolling restart can minimize the impact or do not affect service running.

Restarting Ranger will affect the permissions of all components controlled by Ranger and may affect the normal running of services. Therefore, restart Ranger when the cluster is idle or during off-peak hours. Before the Ranger component is restarted, the policies in the Ranger component still take effect.

Figure 4-7 Restarting a service

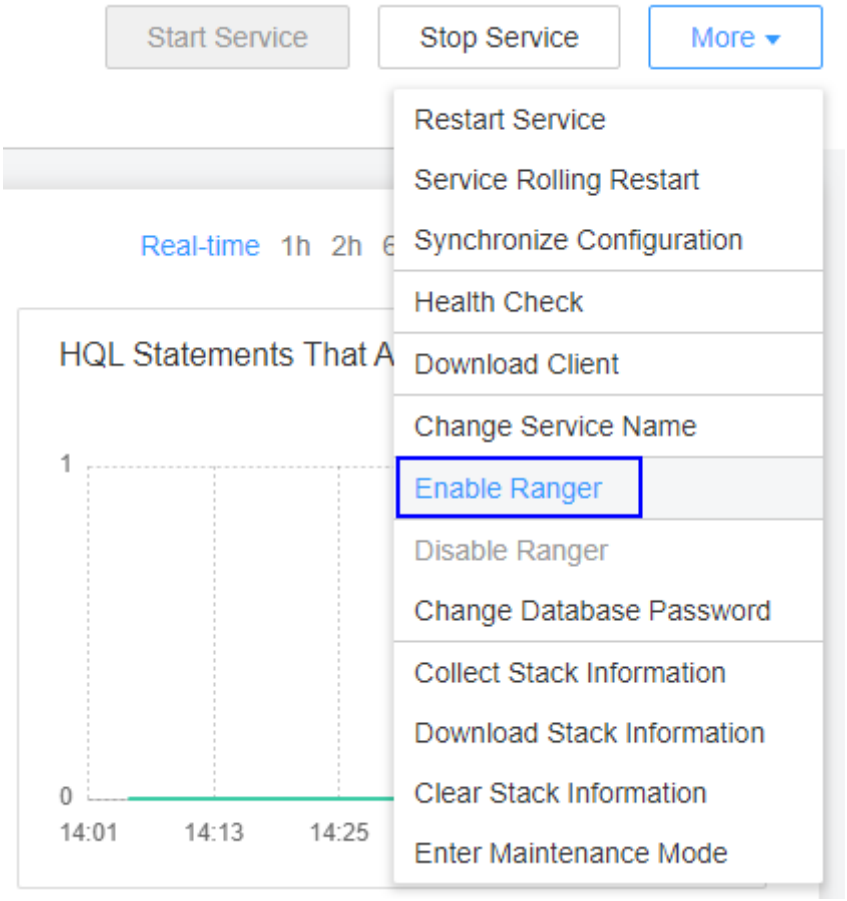


Step 10 Enable Ranger authentication for the component to be authenticated. The Hive component is used as an example.

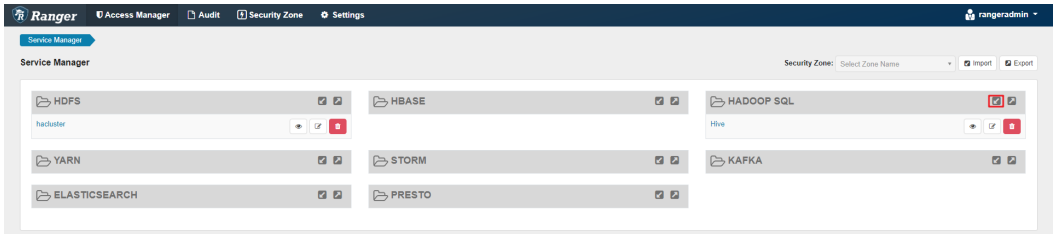
Currently, the following components in an MRS 3.1.*x* cluster support Ranger authentication: HDFS, HBase, Hive, Spark, Impala, Storm, and Kafka.

1. Log in to FusionInsight Manager and choose **Cluster** > **Services** > *Service Name*.
2. In the upper right corner of the **Dashboard** page, click **More** and select **Enable Ranger**.

Figure 4-8 Enabling Ranger authentication



Step 11 Log in to the Ranger web UI and click the import button in the row of the Hive component.

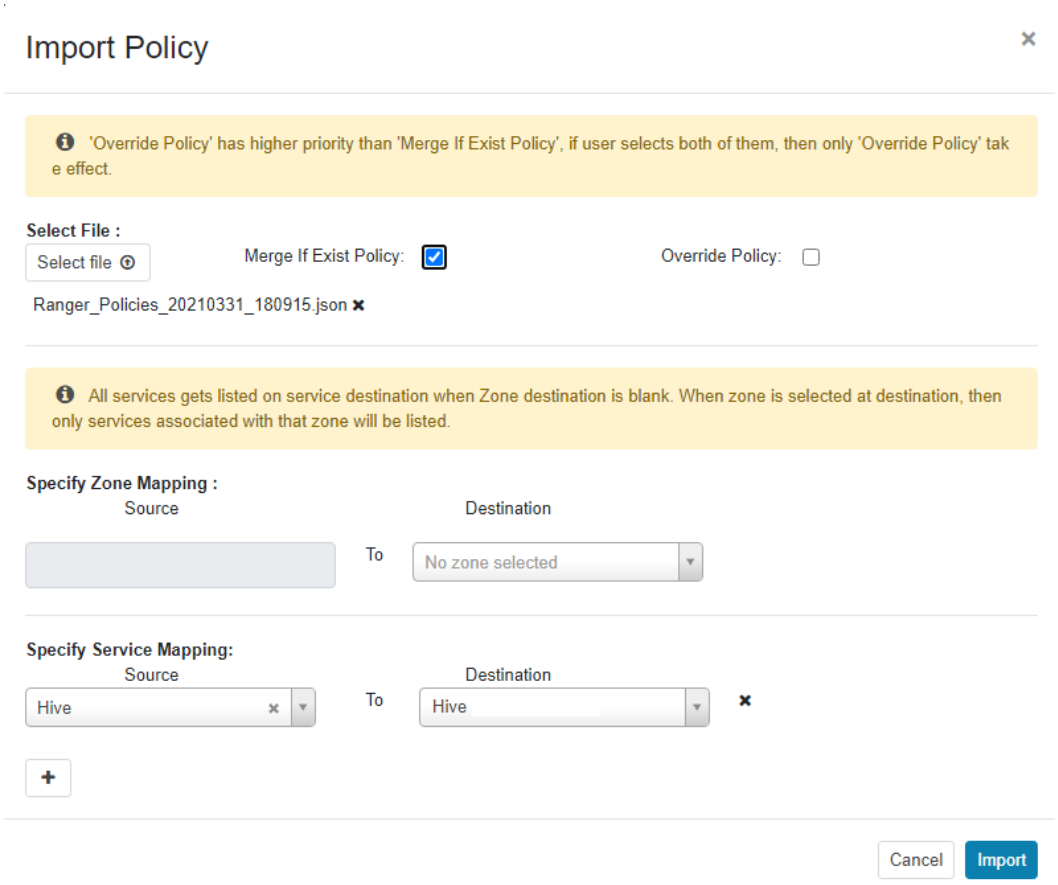


Step 12 Import parameters.

- Click **Select file** and select the authentication policy file downloaded in **Step 3.6**.

- Select **Merge If Exist Policy**.

Figure 4-9 Importing authentication policies



Step 13 Restart the component for which Ranger authentication is enabled.

1. Log in to FusionInsight Manager.
2. Choose **Cluster** > **Services** > **Hive** to go to the Hive service overview page.
3. Choose **More** > **Restart Service** or **More** > **Service Rolling Restart**.

   If you choose **Restart Service**, services will be interrupted during the restart. If you select **Service Rolling Restart**, rolling restart can minimize the impact or do not affect service running.

----End

4.8.3 Configuring a Hive Data Connection

This section describes how to switch the Hive metadata of an active cluster to the metadata stored in a local database or RDS database after you create a cluster. This operation enables multiple MRS clusters to share the same metadata, and the metadata will not be deleted when the clusters are deleted. In this way, Hive metadata migration is not required during cluster migration.

NOTE

- When Hive metadata is switched between different clusters, MRS synchronizes only the permissions in the metadata database of the Hive component. The permission model on MRS is maintained on MRS Manager. Therefore, when Hive metadata is switched between clusters, the permissions of users or user groups cannot be automatically synchronized to MRS Manager of another cluster.
- For clusters whose version is earlier than MRS 3.*x*, if the selected data connection is **RDS MySQL database**, ensure that the database user is **root**. If the user is not **root**, create a user and grant permissions to the user by referring to **Performing Operations Before Data Connection**.
- For clusters whose version is MRS 3.*x* or later, if the selected data connection is **RDS MySQL database**, the database user cannot be user **root**. In this case, create a user and grant permissions to the user by following the instructions provided in **Performing Operations Before Data Connection**.

Configuring a Hive Data Connection

This function is not supported in MRS 3.0.5.

Step 1 Log in to the MRS console. In the navigation pane on the left, choose **Clusters** > **Active Clusters**.

Step 2 Click the name of a cluster to go to the cluster details page.

Step 3 On the **Dashboard** tab page, click **Manage** next to **Data Connection**.

Step 4 On the **Data Connection** dialog box, the data connections associated with the cluster are displayed. You can click **Edit** or **Delete** to edit or delete the data connections.

Step 5 If there is no associated data connection on the **Data Connection** dialog box, click **Configure Data Connection** to add a connection.

NOTE

Only one data connection can be configured for a module type. For example, after a data connection is configured for Hive metadata, no other data connection can be configured for it. If no module type is available, the **Configure Data Connection** button is unavailable.

Table 4-15 Configuring a Hive data connection

Parameter	Description
Component	Hive
Module Type	Hive metadata
Data Connection Type	• RDS PostgreSQL database (supported for clusters of MRS 1.9.*x*) • RDS MySQL database • Local database

Parameter	Description
Instance	This parameter is valid only when **Data Connection Type** is set to **RDS PostgreSQL database** or **RDS MySQL database**. Select the name of the connection between the MRS cluster and the RDS database. This instance must be created before being referenced here. You can click **Create Data Connection** to create a data connection. For details, see **Creating a Data Connection**.

Step 6 Click **Test** to test connectivity of the data connection.

Step 7 After the data connection is successful, click **OK**.

NOTE

- After Hive metadata is configured, restart Hive. Hive will create necessary database tables in the specified database. (If tables already exist, they will not be created.)
- Before restarting the Hive service, ensure that the driver package has been installed on all nodes where Metastore instances are located.
  - Postgres: Use the open source Postgres driver package to replace the existing one of the cluster. Upload the Postgres driver package **postgresql-42.2.5.jar** to the *${BIGDATA_HOME}*/**third_lib/Hive** directory on all MetaStore instance nodes.
  - MySQL: Go to the MySQL official website (https://www.mysql.com/). Choose **DOWNLOADS** and click **MySQL Community (GPL) Downloads**. On the displayed page, click **Connector/J** to download the driver package of the corresponding version and upload the driver package to the **/opt/Bigdata/FusionInsight_HD_*/install/FusionInsight-Hive-*/hive-*/lib/** directory on all RDSMetastore nodes.

----End

4.9 Installing the Third-Party Software Using Bootstrap Actions

4.9.1 Introduction to Bootstrap Actions

Bootstrap actions indicate that you can run your scripts on a specified cluster node before or after starting big data components. You can run bootstrap actions to install additional third-party software, modify the cluster running environment, and perform other customizations.

If you choose to run bootstrap actions when scaling out a cluster, the bootstrap actions will be run on the newly added nodes in the same way. If auto scaling is enabled in a cluster, you can add an automation script in addition to configuring a resource plan. Then the automation script executes the corresponding script on the nodes that are scaled out or in to implement custom operations.

MRS runs the script you specify as user **root**. You can run the **su - *XXX*** command in the script to switch the user.

NOTE

The bootstrap action scripts must be executed as user **root**. Improper use of the script may affect the cluster availability. Therefore, exercise caution when performing this operation.

MRS determines the result based on the return code after the execution of the bootstrap action script. If the return code is **0**, the script is executed successfully. If the return code is not **0**, the execution fails. If a bootstrap action script fails to be executed on a node, the corresponding boot script will fail to be executed. In this case, you can set **Action upon Failure** to choose whether to continue to execute the subsequent scripts. Example 1: If you set **Action upon Failure** to **Continue** for all scripts during cluster creation, all the scripts will be executed regardless of whether they are successfully executed, and the startup process will be complete. Example 2: If a script fails to be executed and **Action upon Failure** is set to **Stop**, subsequent scripts will not be executed and cluster creation or scale-out will fail.

You can add a maximum of 18 bootstrap actions, which will be executed before or after the cluster component is started in the order you specified. The bootstrap actions performed before or after the component startup must be completed within 60 minutes. Otherwise, the cluster creation or scale-out will fail.

4.9.2 Preparing the Bootstrap Action Script

Currently, bootstrap actions support Linux shell scripts only. Script files must end with **.sh**.

Uploading the Installation Packages and Files to an OBS File System

Before compiling a script, you need to upload all required installation packages, configuration packages, and relevant files to the OBS file system in the same region. Because networks of different regions are isolated from each other, MRS VMs cannot download OBS files from other regions.

Compiling a Script for Downloading Files from the OBS File System

You can specify the file to be downloaded from OBS in the script. If you upload files to a private file system, you need to run the **hadoop fs** command to download the files. The following example shows that the **obs://yourbucket/myfile.tar.gz** file will be downloaded to the local host and decompressed to the **/your-dir** directory.

```
#!/bin/bash
source /opt/Bigdata/client/bigdata_env;hadoop fs -D fs.obs.endpoint=<obs-endpoint> -D fs.obs.access.key=<your-ak> -D fs.obs.secret.key=<your-sk> -copyToLocal obs://yourbucket/myfile.tar.gz ./
mkdir -p /<your-dir>
tar -zxvf myfile.tar.gz -C /<your-dir>
```

NOTE

- In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.
- The Hadoop client has been preinstalled on the MRS node. You can run the **hadoop fs** command to download or upload data from or to OBS.
- Obtain the obs-endpoint of each region..

Uploading the Script to the OBS File System

After script compilation, upload the script to the OBS file system in the same region. At the time you specify, each node in the cluster downloads the script from OBS and executes the script as user **root**.

4.9.3 View Execution Records

You can view the execution result of the bootstrap operation on the **Bootstrap Action** page.

Viewing the Execution Result

1. Log in to the MRS console.
2. In the left navigation pane, choose **Clusters** > **Active Clusters**. Click a cluster you want to query.

   The cluster details page is displayed.
3. On the cluster details page, click the **Bootstrap Action** tab. Information about the bootstrap actions added during cluster creation is displayed.

   NOTE

   - You select **Before initial component start** or **After initial component start** in the upper right corner to query information about the related bootstrap actions.
   - The last execution result is listed here. For a newly created cluster, the records of bootstrap actions executed during cluster creation are listed. If a cluster is expanded, the records of bootstrap actions executed on the newly added nodes are listed.

Viewing Execution Logs

If you want to view the run logs of a bootstrap action, set **Action upon Failure** to **Continue** when adding the bootstrap action. And then, log in to each node to view the run logs in the **/var/log/Bootstrap** directory. If you add bootstrap actions before and after component start, you can distinguish bootstrap action logs of the two phases based on the timestamps.

You are advised to print logs in detail in the script so that you can view the detailed run result. MRS redirects the standard output and error output of the script to the log directory of the bootstrap action.

4.9.4 Adding a Bootstrap Action

This operation applies to clusters of MRS 3.*x* or earlier.

In MRS 3.*x*, bootstrap actions cannot be added during cluster creation.

Adding a Bootstrap Action When Creating a Cluster

Step 1 Log in to the MRS console.

Step 2 Click **Create Cluster**. The page for creating a cluster is displayed.

Step 3 Click the **Custom Config** tab.

Step 4 Configure the cluster software and hardware by referring to **Creating a Custom Cluster**.

Step 5 On the **Set Advanced Options** tab page, click **Add** in the **Bootstrap Action** area.

Table 4-16 Parameters

Parameter	Description
Name	Name of a bootstrap action script The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and must not start with a space. The value can contain 1 to 64 characters. **NOTE** A name must be unique in the same cluster. You can set the same name for different clusters.
Script Path	Script path. The value can be an OBS file system path or a local VM path. • An OBS file system path must start with **s3a://** and end with **.sh**, for example, **s3a://mrs-samples/***xxx***.sh**. • A local VM path must start with a slash (/) and end with **.sh**.
Parameters	Bootstrap action script parameters
Execution Node	Select a type of the node where the bootstrap action script is executed.
Executed	Select the time when the bootstrap action script is executed. • Before initial component start • After initial component start
Action upon Failure	Whether to continue to execute subsequent scripts and create a cluster after the script fails to be executed. **NOTE** You are advised to set this parameter to **Continue** in the debugging phase so that the cluster can continue to be installed and started no matter whether the bootstrap action is successful.

Step 6 Click **OK**.

After the bootstrap action is successfully added, you can edit, clone, or delete it in the **Operation** column.

----End

Adding an Automation Script on the Auto Scaling Page

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name. The cluster details page is displayed.

Step 3 On the **Nodes** tab page, click **Auto Scaling** in the **Operation** column of the Task node group. The **Auto Scaling** page is displayed.

If no Task node exists in the cluster, click **Configure Task Node** to add a Task node and then perform this step.

NOTE

For MRS 3.*x* or later, **Configure Task Node** applies only to analysis clusters, streaming clusters, and hybrid clusters.

Step 4 Configure a resource plan.

Configuration procedure:

1. On the **Auto Scaling** page, enable **Auto Scaling**.
2. For example, the **Default Range** of node quantity is set to **2-2**, indicating that the number of Task nodes is fixed to 2 except the time range specified in the resource plan.
3. Click **Configure Node Range for Specific Time Range** under **Default Range**.
4. Configure the **Time Range** and **Node Range** parameters. For example, set **Time Range** to **07:00-13:00**, and **Node Range** to **5-5**. This indicates that the number of Task nodes is fixed to 5 in the time range specified in the resource plan. For details about the parameters, see **Table 4-11**.

   You can click **Configure Node Range for Specific Time Range** to configure multiple resource plans.

Step 5 (Optional) Configure automation scripts.

1. Set **Advanced Settings** to **Configure**.
2. Click **Create**. The **Automation Script** page is displayed.
3. Set the following parameters: **Name**, **Script Path**, **Execution Node**, **Parameter**, **Executed**, and **Action upon Failure**. For details about the parameters, see **Table 4-12**.
4. Click **OK** to save the automation script configurations.

Step 6 Select **I agree to authorize MRS to scale out or scale in nodes based on the above rule**.

Step 7 Click **OK**.

----End

4.10 Viewing Failed MRS Tasks

This section describes how to view and delete a failed MRS task.

Background

If a cluster fails to be created, terminated, scaled out, or scaled in, the **Manage Failed Tasks** page is displayed. Only the tasks that fail to be deleted are displayed on the **Cluster History** page. You can delete a failed task that is not required.

Procedure

Step 1 Log in to the MRS console.

Step 2 In the left navigation pane, choose **Clusters** > **Active Clusters**.

Step 3 Click  or the number on the right of **Failed Tasks**. The **Manage Failed Tasks** page is displayed.

Step 4 In the **Operation** column of the cluster that you want to start, click **Delete**.

In this step, only one job can be deleted.

Step 5 You can click **Delete All** in the upper left corner of the task list to delete all failed tasks.

----End

4.11 Viewing Information of a Historical Cluster

Choose **Clusters > Cluster History** and click the name of a target cluster. You can view the cluster configuration and deployed node information.

The following table describes the parameters for the historical cluster information.

Table 4-17 Basic cluster information

Parameter	Description
Cluster Name	Name of a cluster. The cluster name is set when the cluster is created.
Cluster Status	Status of a cluster.
Cluster Version	Cluster version
Cluster Type	Type of the cluster to be created.
Obtaining a cluster ID	Unique identifier of a cluster, which is automatically assigned when a cluster is created
Created	Time when a cluster is created.
AZ	Availability zone (AZ) in the region of a cluster, which is set when a cluster is created.
Default Subnet	Subnet selected during cluster creation. A subnet provides dedicated network resources that are isolated from other networks, improving network security.
VPC	VPC selected during cluster creation. A VPC is a secure, isolated, and logical network environment.

Parameter	Description
OBS Permission Control	Click **Manage** and modify the mapping between MRS users and OBS permissions. For details, see **Configuring Fine-Grained Permissions for MRS Multi-User Access to OBS**.
Creating a data connection	Click **Manage** to view the data connection type associated with the cluster. For details, see **Configuring Data Connections**.
Agency	Click **Manage Agency** to bind or modify an agency for the cluster. An agency allows ECS or BMS to manage MRS resources. You can configure an agency of the ECS type to automatically obtain the AK/SK to access OBS. For details, see **Configuring a Storage-Compute Decoupled Cluster (Agency)**. The **MRS_ECS_DEFAULT_AGENCY** agency has the OBSOperateAccess permission of OBS and the CESFullAccess (for users who have enabled fine-grained policies), CES Administrator, and KMS Administrator permissions in the region where the cluster is located.
Key Pair	Name of a key pair. Set this parameter when creating a cluster. If the login mode is set to password during cluster creation, this parameter is not displayed.
Kerberos Authentication	Whether to enable Kerberos authentication when logging in to Manager.
Enterprise project	Enterprise project to which a cluster belongs. Only on the **Active Clusters** page, you can click the name of an enterprise project to go to its **Enterprise Project Management** page.
Security Group	Security group name of the cluster.
Streaming Core Node LVM	Indicates whether to enable the Logical Volume Manager (LVM) function of streaming Core nodes.
Data Disk Key Name	Name of the key used to encrypt data disks. To manage the used keys, log in to the key management console.
Data Disk Key ID	ID of the key used to encrypt data disks.
Component Version	Version of each component installed in the cluster.
License Version	License version of the cluster.
Agency	Delegates ECSs or BMSs to manage some of your resources.

Go back to the historical clusters page. You can use the following buttons to perform operations. For details about the buttons, see the following table.

Table 4-18 Icon description

Icon	Description
	Click  to manually refresh the node information.
	Enter a cluster name in the search bar and click  to search for a cluster.

5 Managing Clusters

5.1 Logging In to a Cluster

5.1.1 MRS Cluster Node Overview

This section describes remote login, MRS cluster node types, and node functions.

MRS cluster nodes support remote login. The following remote login methods are available:

- GUI login: Use the remote login function provided by the ECS management console to log in to the Linux interface of the Master node in the cluster.
- SSH login: Applies to Linux ECSs only. You can use a remote login tool (such as PuTTY) to log in to an ECS. The ECS must have a bound EIP.

  For details about how to apply for and bind EIP for the Master node, see **Virtual Private Cloud** > **User Guide** > **Elastic IP** > **Assigning an EIP and Binding It to an ECS**.

  You can log in to a Linux ECS using either a key pair or password.

  NOTICE

  If you use a key pair to access a node in a cluster, you need to log in to the node as user **root**. For details, see **Logging In to an ECS Using a Key Pair (SSH)**.

  For details about how to access a cluster node using a password, see **Logging In to an ECS Using a Password (SSH)**.

In an MRS cluster, a node is an ECS. **Table 5-1** describes the node types and node functions.

Table 5-1 Cluster node types

Node Type	Functions
Master node	Management node of an MRS cluster. It manages and monitors the cluster. In the navigation tree of the MRS management console, choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page. On the **Nodes** tab page, view the **Name**. The node that contains **master1** in its name is the Master1 node. The node that contains **master2** in its name is the Master2 node. You can log in to a Master node either using VNC on the ECS management console or using SSH. After logging in to the Master node, you can access Core nodes without entering passwords. The system automatically deploys the Master nodes in active/standby mode and supports the high availability (HA) feature for MRS cluster management. If the active management node fails, the standby management node switches to the active state and takes over services. To determine whether the Master1 node is the active management node, see **Determining Active and Standby Management Nodes of Manager**.
Core node	Work node of an MRS cluster. It processes and analyzes data and stores process data.
Task node	Compute node. It is used for auto scaling when the computing resources in a cluster are insufficient.

5.1.2 Logging In to an ECS

This section describes how to remotely log in to an ECS in an MRS cluster using the remote login (VNC mode) function provided on the ECS management console or a key or password (SSH mode). Remote login (VNC mode) is mainly used for emergency O&M. In other scenarios, it is recommended that you log in to the ECS using SSH.

NOTE

To log in to a cluster node using SSH, you need to manually add an inbound rule in the security group of the cluster. The source address is **Client IPv4 address/32** (or **Client IPv6 address/128**) and the port number is **22**. For details, see **Virtual Private Cloud > User Guide > Security > Security Group > Adding a Security Group Rule**.

Logging In to an ECS Using VNC

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 On the **Nodes** tab page, click the name of a Master node in the Master node group to log in to the ECS management console.

Step 4 In the upper right corner, click **Remote Login**.

----End

Logging In to an ECS Using a Key Pair (SSH)

Logging In to the ECS from Local Windows

To log in to the Linux ECS from local Windows, perform the operations described in this section. The following procedure uses PuTTY as an example to log in to the ECS.

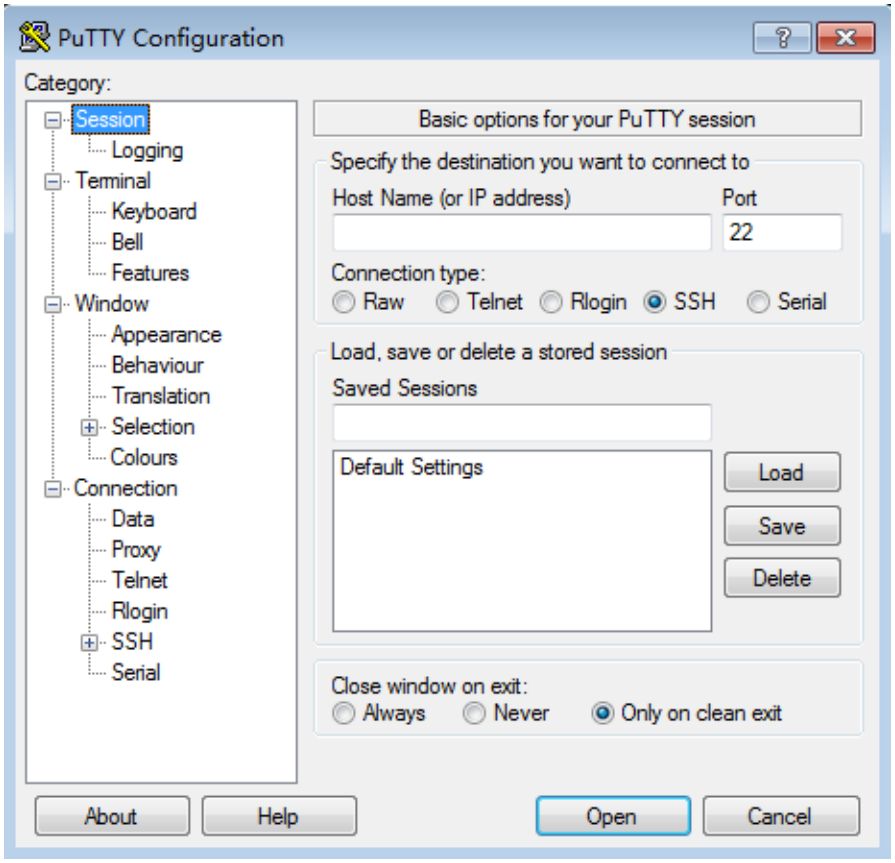
1. Log in to the MRS management console.
2. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.
3. On the **Nodes** tab page, click the name of a Master node in the Master node group to log in to the ECS management console.
4. Click the **EIPs** tab, click **Bind EIP** to bind an EIP to the ECS, and record the EIP. If an EIP has been bound to the ECS, skip this step.
5. Check whether the private key file has been converted to **.ppk** format.
   - If yes, go to **10**.
   - If no, go to **6**.
6. Run PuTTY.
7. In the **Actions** area, click **Load** and import the private key file you used during ECS creation.

   Ensure that the private key file is in the format of **All files (*.*)**.
8. Click **Save private key**.
9. Save the converted private key, for example, **kp-123.ppk**, to a local directory.
10. Run PuTTY.
11. Choose **Connection** > **Data**. Enter the image username in **Auto-login username**.

    NOTE

    The image username for cluster nodes is **root**.
12. Choose **Connection** > **SSH** > **Auth**. In the last configuration item **Private key file for authentication**, click **Browse** and select the private key converted in **9**.
13. Click **Session**.
    a. **Host Name (or IP address)**: Enter the EIP bound to the ECS.
    b. **Port**: Enter **22**.
    c. **Connection Type**: Select **SSH**.
    d. **Saved Sessions**: Task name, which can be clicked for remote connection when you use PuTTY next time

Figure 5-1 Clicking **Session**



14. Click **Open** to log in to the ECS.

    If you log in to the ECS for the first time, PuTTY displays a security warning dialog box, asking you whether to accept the ECS security certificate. Click **Yes** to save the certificate to your local registry.

Logging In to the ECS from Local Linux

To log in to the Linux ECS from local Linux, perform the operations described in this section. The following procedure uses private key file **kp-123.pem** as an example to log in to the ECS. The name of your private key file may differ.

1. On the Linux CLI, run the following command to change operation permissions:

   chmod 400 /*path*/kp-123.pem

   NOTE

   In the preceding command, *path* refers to the path where the key file is saved.

2. Run the following command to log in to the ECS:

   ssh -i /*path*/kp-123.pem*Default username*@*EIP*

   For example, if the default username is **root** and the EIP is **123.123.123.123**, run the following command:

   ssh -i /*path*/kp-123.pem root@123.123.123.123

NOTE

- *path* indicates the path where the key file is saved.
- *EIP* indicates the EIP bound to the ECS.
- The image username is **root** for cluster nodes.

Logging In to an ECS Using a Password (SSH)

Logging In to the ECS from Local Windows

To log in to the Linux ECS from local Windows, perform the operations described in this section. The following procedure uses PuTTY as an example to log in to the ECS.

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 On the **Nodes** tab page, click the name of a Master node in the Master node group to log in to the ECS management console.

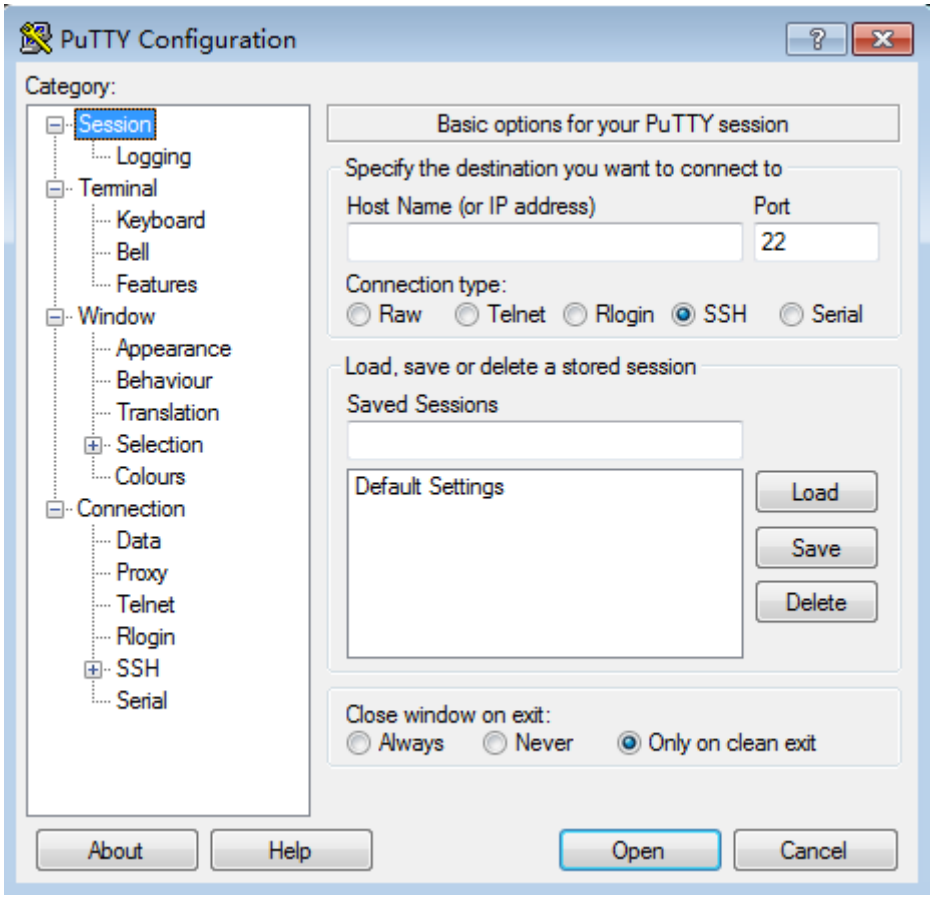
Step 4 Click the **EIPs** tab, click **Bind EIP** to bind an EIP to the ECS, and record the EIP. If an EIP has been bound to the ECS, skip this step.

Step 5 Run PuTTY.

Step 6 Click **Session**.

1. **Host Name (or IP address)**: Enter the EIP bound to the ECS.
2. **Port**: Enter **22**.
3. **Connection Type**: Select **SSH**.
4. **Saved Sessions**: Task name, which can be clicked for remote connection when you use PuTTY next time

Figure 5-2 Clicking **Session**



Step 7 Click **Window** and select **UTF-8** for **Remote character set:** in **Translation**.

Step 8 Click **Open** to log in to the ECS.

If you log in to the ECS for the first time, PuTTY displays a security warning dialog box, asking you whether to accept the ECS security certificate. Click **Yes** to save the certificate to your local registry.

Step 9 After the SSH connection to the ECS is set up, enter the username and password as prompted to log in to the ECS.

NOTE

The username is **root** and the password is the one you set during cluster creation.

----End

Logging In to the ECS from Local Linux

If the local host runs Linux, perform steps **Step 1** to **Step 4** to bind an EIP to the ECS, and run the following command on the CLI to log in to the ECS: **ssh *EIP bound by the ECS***

5.1.3 Determining Active and Standby Management Nodes of Manager

This section describes how to determine the active and standby management nodes of Manager on the Master1 node.

Background

You can log in to other nodes in the cluster from the Master node. After logging in to the Master node, you can determine the active and standby management nodes of Manager and run commands on corresponding management nodes.

In active/standby mode, a switchover can be implemented between Master1 and Master2. For this reason, Master1 may not be the active management node for Manager.

Procedure

Step 1 Confirm the Master nodes of an MRS cluster.

1. In the navigation tree of the MRS management console, choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page. View basic information of the specified cluster.
2. On the **Nodes** tab page, view the node name. The node that contains **master1** in its name is the Master1 node. The node that contains **master2** in its name is the Master2 node.

Step 2 Determine the active and standby Manager management nodes.

1. Remotely log in to the Master1 node. For details, see **Logging In to an ECS**.

   Master nodes support Cloud-Init. The preset username for Cloud-Init is **root** and the password is the one you set during cluster creation.
2. Run the following commands to switch the user:

   sudo su - root

   su - omm
3. Run the following command to identify the active and standby management nodes:

   For versions earlier than MRS 3.*x*, run the **sh ${BIGDATA_HOME}/om-0.0.1/sbin/status-oms.sh** command.

   For MRS 3.*x* or later: Run the **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** command.

   In the command output, the node whose **HAActive** is **active** is the active management node (mgtomsdat-sh-3-01-1 in the following example), and the node whose **HAActive** is **standby** is the standby management node (mgtomsdat-sh-3-01-2 in the following example).

   ```
   Ha mode
   double
   NodeName            HostName                HAVersion       StartTime           HAActive
   HAAllResOK          HARunPhase
   192-168-0-30        mgtomsdat-sh-3-01-1     V100R001C01     2014-11-18 23:43:02
   active              normal                  Actived
   192-168-0-24        mgtomsdat-sh-3-01-2     V100R001C01     2014-11-21 07:14:02
   standby             normal                  Deactived
   ```

   NOTE

   If the Master1 node to which you have logged in is the standby management node and you need to log in to the active management node, run the following command:

   ssh *IP address of Master2 node*

----End

5.2 Cluster Overview

5.2.1 Cluster List

You can quickly view the status of all clusters and jobs by viewing the dashboard information, and obtain relevant MRS documents from **Help** in the left navigation pane on the MRS console.

MRS is used to manage and analyze massive data. It is easy to use. You can create a cluster and add MapReduce, Spark, and Hive jobs to the cluster to analyze and process user data. After being processed, you can transmit the data in SSL encryption mode to OBS to ensure data integrity and confidentiality.

Cluster Status

Table 5-2 lists the statuses of all MRS clusters after you log in to the MRS management console.

Table 5-2 Cluster status

Status	Description
Starting	If a cluster is being created, the cluster is in the **Starting** state.
Running	If a cluster is created successfully and all components in the cluster are normal, the cluster is in the **Running** state.
Scaling out	If the Core or Task node in a cluster is being added, the cluster is in the **Scaling out** state. **NOTE** If the cluster scale-out fails, you can add node to the cluster again.
Scaling in	If you stop, delete, change or reinstall the OSs of cluster nodes, and modify the specifications of the cluster node, the cluster nodes are being terminated. Then, the cluster is in the **Scaling in** state.
Abnormal	If some components in a cluster are abnormal, the cluster is **Abnormal**.
Terminating	If a cluster node is being terminated, the cluster is in the **Terminating** state.
Terminated	The cluster has been terminated. This parameter is displayed only in **Cluster History**.

Job Status

Table 5-3 describes the status of jobs that you execute after logging in to the MRS management console.

Table 5-3 Job status

Status	Description
Accepted	Initial status of a job after it is successfully submitted.
Running	A job is being executed.
Completed	A job has been executed and completed successfully.
Terminated	A job is stopped during execution.
Abnormal	An error occurs during job execution or job execution fails.

5.2.2 Checking the Cluster Status

The cluster list contains all clusters in MRS. You can view clusters in various states. If a large number of clusters are involved, navigate through multiple pages to view all of the clusters.

MRS, as a platform managing and analyzing massive data, provides a PB-level data processing capability. MRS allows you to multiple clusters. The cluster quantity is subject to that of ECSs.

Clusters are listed in chronological order by default in the cluster list, with the most recent cluster displayed at the top. **Table 5-4** describes the cluster list parameters.

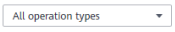
- **Active Clusters**: contain all clusters except the clusters in the **Failed** and **Terminated** states.
- **Cluster History**: contains the tasks in the **Terminated** states. Only clusters terminated within the last six months are displayed. If you want to view clusters terminated six months ago, contact technical support engineers.
- **Failed Tasks**: only contain the tasks in the **Failed** state. Task failures include:
  - Cluster creation failure
  - Cluster termination failure
  - Cluster scale-out failure
  - Cluster scale-in failure
  - Cluster patch installation failure (supported only by versions earlier than MRS 3.x)
  - Cluster patch uninstallation failure (supported only by versions earlier than MRS 3.x)
  - Cluster specifications upgrade failure

Table 5-4 Parameters in the active cluster list

Parameter	Description
Name/ID	Cluster name, which is set when a cluster is created. Unique identifier of a cluster, which is automatically assigned when a cluster is created. • : Change the cluster name. • : Copy the cluster ID.
Cluster Version	Cluster version.
Nodes	Number of nodes that can be deployed in a cluster. This parameter is set when a cluster is created.
Status	Status and operation progress description of a cluster. The cluster creation progress includes: • Verifying cluster parameters • Applying for cluster resources • Creating VMs • Initializing VMs • Installing MRS Manager • Deploying the cluster • Cluster installation failed The cluster scale-out progress includes: • Preparing for scale-out • Creating VMs • Initializing VMs • Adding nodes to the cluster • Scale-out failed The cluster scale-in progress includes: • Preparing for scale-in • Decommissioning instance • Deleting VMs • Deleting nodes from the cluster • Scale-in failed The system will display causes of cluster installation, scale-out, and scale-in failures. For details, see **Table 4-5**.
Created	The cluster node is successfully created.
Terminated	Time when a cluster node stops and the cluster node begins to be terminated. This parameter is valid only for historical clusters displayed on the **Cluster History** page.

Parameter	Description
AZ	Availability zone (AZ) in the region of a cluster, which is set when a cluster is created.
Enterprise Project	Enterprise project to which a cluster belongs.
Operation	**Terminate**: If you want to terminate a cluster after jobs are complete, click **Terminate**. The cluster status changes from **Running** to **Terminating**. After the cluster is terminated, the cluster status will change to **Terminated** and will be displayed in **Cluster History**. If the MRS cluster fails to be deployed, the cluster is automatically terminated. This parameter is displayed in **Active Clusters** only. **NOTE** Typically after data is analyzed and stored, or when the cluster encounters an exception and cannot work, you can terminate a cluster. If a cluster is terminated before data processing and analysis are completed, data loss may occur. Therefore, exercise caution when terminating a cluster.

Table 5-5 Button description

Button	Description
	Select an enterprise project from the drop-down list to filter the corresponding cluster.
	In the drop-down list, select a status to filter clusters: • Active Clusters – All operation types: displays all existing clusters. – Starting: displays existing clusters in the **Starting** state. – Running: displays existing clusters in the **Running** state. – Scaling out: displays existing clusters in the **Scaling out** state. – Scaling in: displays existing clusters in the **Scaling in** state. – Abnormal: displays existing clusters in the **Abnormal** state. – Terminating: displays existing clusters in the **Terminating** state.
	Choose **Clusters** > **Active Clusters** and click  to go to the page for managing failed tasks.  *Num*: displays the failed tasks in the **failed** state.

Button	Description
	Enter a cluster name in the search bar and click  to search for a cluster.
Search by Tag	Click **Search by Tag**, enter the tag of the cluster to be queried, and click **Search** to search for the clusters. You can select a tag key or tag value from their drop-down lists. When the tag key or tag value is exactly matched, the system can automatically locate the target cluster. If you enter multiple tags, their intersections are used to search for the cluster.
	Click  to manually refresh the cluster list.

5.2.3 Viewing Basic Cluster Information

You can monitor and manage the clusters you have created. Choose **Clusters > Active Clusters**. Select a cluster and click its name to go to the cluster details page. On the displayed page, view the basic configuration and node information of the cluster.

NOTE

On the MRS console, operations performed on an ECS cluster are basically the same as those performed on a BMS cluster. This document describes operations on an ECS cluster. If operations on the two clusters differ, the operations will be described separately.

On the cluster details page, click **Dashboard**. Table 5-6 describes the parameters on the **Dashboard** tab page.

Table 5-6 Basic cluster information

Parameter	Description
Cluster Name	Name of a cluster. Set this parameter when creating a cluster. Click  to change the cluster name. For versions earlier than MRS 3.x, only the cluster name displayed on the MRS management console is changed, while the cluster name on MRS Manager is not changed synchronously.
Cluster Status	Cluster status. For details, see Table 5-2.

Parameter	Description
MRS Manager	Portal for the Manager page. • For MRS 3.*x* or later, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. • For versions earlier than MRS 3.x, you need to bind an EIP and add a security group rule as prompted before accessing the MRS Manager page. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.
Cluster Version	MRS version information.
Cluster Type	There are three types of clusters: • **Analysis cluster**: is used for offline data analysis and provides Hadoop components. • **Streaming cluster**: is used for streaming tasks and provides stream processing components. • **Hybrid cluster**: is used for both offline data analysis and streaming processing and provides Hadoop components and streaming processing components. • Custom: An MRS cluster with all custom components. MRS 3.*x* and later versions support this type.
Cluster ID	Unique identifier of a cluster, which is automatically assigned when a cluster is created.
Created	Time when a cluster is created.
AZ	Availability zone (AZ) in the region of a cluster, which is set when a cluster is created.
Default Subnet	Subnet selected during cluster creation. If the subnet IP addresses are insufficient, click **Change Subnet** to switch to another subnet in the same VPC of the current cluster to obtain more available subnet IP addresses. Changing a subnet does not affect the IP addresses and subnets of existing nodes. A subnet provides dedicated network resources that are isolated from other networks, improving network security.
VPC	VPC selected during cluster creation. A VPC is a secure, isolated, and logical network environment.
Elastic IP (EIP)	After binding an EIP to an MRS cluster, you can use the EIP to access the Manager web UI of the cluster.
OBS Permission Control	Click **Manage** and modify the mapping between MRS users and OBS permissions. For details, see **Configuring Fine-Grained Permissions for MRS Multi-User Access to OBS**.
Data Connection	Click **Manage** to view the data connection type associated with the cluster. For details, see **Configuring Data Connections**.

Parameter	Description
Agency	Click **Manage Agency** to bind or modify an agency for the cluster. An agency allows ECS or BMS to manage MRS resources. You can configure an agency of the ECS type to automatically obtain the AK/SK to access OBS. For details, see **Configuring a Storage-Compute Decoupled Cluster (Agency)**. The **MRS_ECS_DEFAULT_AGENCY** agency has the OBS OperateAccess permission of OBS and the CES FullAccess (for users who have enabled fine-grained policies), CES Administrator, and KMS Administrator permissions in the region where the cluster is located.
Key Pair	Name of a key pair. Set this parameter when creating a cluster. If the login mode is set to password during cluster creation, this parameter is not displayed.
Kerberos Authentication	Whether to enable Kerberos authentication when logging in to Manager.
Logging	Used to collect logs about cluster creation and scaling failures.
Enterprise Project	Enterprise project to which a cluster belongs. Only on the **Active Clusters** page, you can click the name of an enterprise project to go to its **Enterprise Project Management** page.
Security Group	Security group name of the cluster.
Streaming Core Node LVM	Indicates whether to enable the Logical Volume Manager (LVM) function of streaming Core nodes.
Data Disk Key Name	Name of the key used to encrypt data disks. To manage the used keys, log in to the key management console.
Data Disk Key ID	ID of the key used to encrypt data disks.
IAM User Synchronization	IAM user information can be synchronized to an MRS cluster for cluster management. For details, see **Synchronizing IAM Users to MRS**. **NOTE** The **Components**, **Tenants**, and **Backups & Restorations** tab pages on the cluster details page can be used only after users are synchronized. After clusters of MRS 3.x are synchronized, you can use the **Component Management** function.
Secure Communications	Used to display the security authorization status. You can click  to enable or disable security authorization. Disabling security authorization brings high risks. Exercise caution when performing this operation. For details, see **Communication Security Authorization**.

Table 5-7 Component versions

Parameter	Description
Hadoop Version	Displays the Hadoop version information.
Spark Version	Version of the Spark component. Only clusters of versions earlier than MRS 3.x support this parameter.
HBase Version	Displays the HBase version information.
Hive Version	Displays the Hive version information.
Hue Version	Displays the Hue version information.
Loader Version	Displays the Loader version information.
Kafka Version	Displays the Kafka version information.
Storm Version	Displays the Storm version information.
Flume Version	Displays the Flume version information.
Tez Version	Displays the Tez version information.
Presto Version	Displays the Presto version information.
KafkaManager Version	Displays the KafkaManager version information.
Flink Version	Displays the Flink version information.
Alluxio Version	Displays the Alluxio version information.
Ranger Version	Displays the Ranger version information.
Impala Version	Displays the Impala version information.
Kudu Version	Displays the Kudu version information.
Spark2x Version	Displays the version information about the Spark2x component. Only clusters of MRS 3.x or later support this function.
Oozie Version	Displays the Oozie version information. Only clusters of MRS 3.x or later support this function.
ClickHouse Version	Displays ClickHouse version information. Only clusters of MRS 3.x or later support this function.

On the cluster details page, click **Nodes**. For details about the node parameters, see **Table 5-8**.

Table 5-8 Node information

Parameter	Description
Configure Task Node	Used to add a Task node. For details, see **Adding a Task Node**. For 3.x and later versions, this operation applies only to the analysis cluster, streaming cluster, and hybrid cluster.
Add Node Group	This parameter applies only to 3.x and later versions. It applies to customized clusters only and is used to add node groups. For details, see **Adding a Node Group**.
Node Group	Node group name.
Node Type	Node type: • **Master**: A Master node in an MRS cluster manages the cluster, assigns MapReduce executable files to Core nodes, traces the execution status of each job, and monitors the DataNode running status. • A task node group is a group of nodes where only data roles that do not store data are deployed. The roles include NodeManager, ThriftServer, Thrift1Server, RESTServer, Supervisor, LogViewer, HBaseIndexer, and TagSync. • If other roles are deployed in the node group in addition to the preceding roles, the node group is the Core node group. On the **Nodes** tab page, click ⌄ next to a node group name to unfold the nodes contained in the node group. Click a node name to remotely log in to the ECS using the password or key pair configured during cluster creation. For details about the parameters, see **Managing Components and Monitoring Hosts**.
Node Count	Number of nodes in a node group.
Operation	• **Scale Out**: For details, see **Manually Scaling Out a Cluster**. • **Scale In**: For details, see **Manually Scaling In a Cluster**. • **Auto Scaling**: For details, see **Configuring an Auto Scaling Rule**. • **View Roles**: You can view information about roles deployed on the node group. This function applies only to custom clusters of 3.x and later.

5.2.4 Viewing Cluster Patch Information

To view patch information about cluster components, you can download the required patch if the cluster component, such as Hadoop or Spark, is faulty. On the MRS console, choose **Clusters > Active Clusters**, select a cluster, and click the

cluster name. On the cluster details page that is displayed, upgrade the component and rectify the fault.

NOTE

MRS 3.x does not have patch version information. Therefore, this section is not involved.

- Patch Name: name of the patch package
- Published: time when the patch package is released
- Status: patch status
- Patch Description: patch version description
- Operation: patch installation or uninstallation

5.2.5 Viewing and Customizing Cluster Monitoring Metrics

MRS cluster nodes are classified into management nodes, control nodes, and data nodes. The change trends of key host monitoring metrics on each type of node can be calculated and displayed as curve charts in reports based on the customized periods. If a host belongs to multiple node types, the metric statistics will be repeatedly collected.

This section provides overview of MRS clusters and describes how to view, customize, and export node monitoring metrics on MRS Manager.

Method 1 (applicable to clusters of versions earlier than MRS 3.x):

Step 1 Choose **Clusters** > **Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click the **Dashboard** tab, you can view the cluster host health status statistics on the lower part of the displayed tab page.

Step 3 To view or export reports of other metrics, click **Access Manager** next to **MRS Manager** in the **Basic Information** area to access the Manager page. For details, see **Accessing Manager**.

Step 4 On the Manager page, view, customize, and export the node monitoring metric report. For details, see **Dashboard**.

----End

Method 2

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters** and click a cluster name to go to the cluster details page.

Step 3 In the **Basic Information** area on the **Dashboard** tab page, click **Click to synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

Step 4 After the synchronization is complete, you can view the cluster monitoring metric report on the right of the page.

Step 5 In time range area, specify a period to view monitoring data. The options are as follows:

- Last 1 hour

- Last 3 hours
- Last 12 hours
- Last 24 hours
- Recent 7 days
- Recent 30 days
- Customize: You can customize the period for viewing monitoring data.

Step 6 Customize a monitoring metric report.

1. Click **Customize** and select monitoring metrics to be displayed.

   MRS supports a maximum of 14 monitoring metrics, but at most 12 customized monitoring metrics can be displayed on the page.

   - Cluster Host Health Status
   - Cluster Network Read Speed Statistics
   - Host Network Read Speed Distribution
   - Host Network Write Speed Distribution
   - Cluster Disk Write Speed Statistics
   - Cluster Disk Usage Statistics
   - Cluster Disk Information
   - Host Disk Usage Statistics
   - Cluster Disk Read Speed Statistics
   - Cluster Memory Usage Statistics
   - Host Memory Usage Distribution
   - Cluster Network Write Speed Statistics
   - Host CPU Usage Distribution
   - Cluster CPU Usage Statistics

2. Click **OK** to save the selected monitoring metrics for display.

   NOTE

   Click **Clear** to cancel all the selected monitoring metrics in a batch.

Step 7 Export a monitoring report.

1. Select a period. The options are as follows:
   - Last 1 hour
   - Last 3 hours
   - Last 12 hours
   - Last 24 hours
   - Recent 7 days
   - Recent 30 days
   - Customize: You can customize the period for viewing monitoring data.
2. Click **Export**. MRS will generate a report about the selected monitoring metrics in a specified time of period. Save the report.

----End

Method 3: (applicable to MRS 3.x clusters)

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters** and click a cluster name to go to the cluster details page.

Step 3 In the **Basic Information** area on the **Dashboard** tab page, click **Click to synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

Step 4 After the synchronization is complete, you can view the cluster monitoring metric report on the right of the page.

Step 5 In time range area, specify a period to view monitoring data. The options are as follows:

- Last 1 hour
- Last 3 hours
- Last 12 hours
- Last 24 hours
- Recent 7 days
- Recent 30 days
- Customize: You can customize the period for viewing monitoring data.

Step 6 Customize a monitoring metric report.

1. Click **Customize** and select monitoring metrics to be displayed.

   At most 12 customized monitoring metrics can be displayed on the page.

2. Click **OK** to save the selected monitoring metrics for display.

   NOTE

   Click **Clear** to cancel all the selected monitoring metrics in a batch.

----End

5.2.6 Managing Components and Monitoring Hosts

You can manage the following status and metrics of all components (including role instances) and hosts on the MRS console:

- Status information: includes operation, health, configuration, and role instance status.
- Indicator information: includes key monitoring indicators for each component.
- Export monitoring metrics. (This function is not supported in MRS 3.x or later.)

NOTE

- For , see **Managing Services and Monitoring Hosts**.
- For MRS 3.x or later, see **Procedure**.
- You can set the interval for automatically refreshing the page or click to refresh the page immediately.
- Component management supports the following parameter values:
  - Refresh every 30 seconds
  - Refresh every 60 seconds
  - Stop refreshing

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Managing Components

NOTE

For details about how to perform operations on MRS Manager, see **Managing Service Monitoring**.

Step 1 On the MRS cluster details page, click **Components**.

On the **Components** tab page, **Service**, **Operating Status**, **Health Status**, **Configuration Status**, **Role**, and **Operation** are displayed in the component list.

- **Table 5-9** describes the service operating status.

  Table 5-9 Service operating status

Status	Description
Started	The service is started.
Stopped	The service is stopped.
Failed to start	Failed to start the role instance.
Failed to stop	Failed to stop the service.
Unknown	Indicates initial service status after the background system restarts.

- **Table 5-10** describes the service health status.

Table 5-10 Service health status

Status	Description
Good	Indicates that all role instances in the service are running properly.
Faulty	Indicates that the running status of at least one role instance is **Faulty** or the status of the service on which the current service depends is abnormal.
Unknown	Indicates that all role instances in the service are in the **Unknown** state.
Restoring	Indicates that the background system is restarting the service.
Partially Healthy	Indicates that the status of the service on which the service depends is abnormal, and APIs related to the abnormal service cannot be invoked by external systems.

- Table 5-11 describes the service health status.

Table 5-11 Service configuration status

Status	Description
Synchronized	The latest configuration takes effect.
Configuration expired	The latest configuration does not take effect after the parameter modification. Related services need to be restarted.
Configuration failed	The communication is incorrect or data cannot be read or written during the parameter configuration. Use **Synchronize Configuration** to rectify the fault.
Configuring	Parameters are being configured.
Unknown	Indicates that configuration status cannot be obtained.

By default, the **Service** column is sorted in ascending order. You can click the icon next to **Service**, **Operating Status**, **Health Status**, or **Configuration Status** to change the sorting mode.

Step 2 Click a specified service in the list to view its status and metric information.

Step 3 Customize and view monitoring graphs.

1. In the **Charts** area, click **Customize** to customize service monitoring metrics.
2. In **Period** area, select a time of period and click **View** to view the monitoring data within the time period.

----End

Managing Role Instances

NOTE

For versions earlier than MRS 3.x, see **Managing Role Instances**.

Step 1 On the MRS cluster details page, click **Components**. In the component list, click the specified service name.

Step 2 Click **Instances** to view the role status.

The role instance list contains the Role, Host Name, Management IP Address, Service IP Address, Rack, Running Status, and Configuration Status of each instance.

- **Table 5-12** shows the running status of a role instance.

Table 5-12 Role instance running status

Status	Description
Good	Indicates that the instance is running properly.
Bad	Indicates that the instance cannot run properly.
Decommissioned	Indicates that the instance is out of service.
Not started	Indicates that the instance is stopped.
Unknown	Indicates that the initial status of the instance cannot be detected.
Starting	Indicates that the instance is being started.
Stopping	Indicates that the instance is being stopped.
Restoring	Indicates that an exception may occur in the instance and the instance is being automatically rectified.
Decommissioning	Indicates that the instance is being decommissioned.
Recommissioning	Indicates that the instance is being recommissioned.
Failed to start	Indicates that the service fails to be started.
Failed to stop	Indicates that the service fails to be stopped.

- **Table 5-13** shows the configuration status of a role instance.

Table 5-13 Role instance configuration status

Status	Description
Synchronized	The latest configuration takes effect.

Status	Description
Configuration expired	The latest configuration does not take effect after the parameter modification. Related services need to be restarted.
Configuration failed	The communication is incorrect or data cannot be read or written during the parameter configuration. Use **Synchronize Configuration** to rectify the fault.
Configuring	Parameters are being configured.
Unknown	Current configuration status cannot be obtained.

By default, the **Role** column is sorted in ascending order. You can click the sorting icon next to **Role**, **Host Name**, **OM IP Address**, **Business IP Address**, **Rack**, **Running Status**, or **Configuration Status** to change the sorting mode.

You can filter out all instances of the same role in the **Role** column.

You can set search criteria in the role search area by clicking **Advanced Search**, and click **Search** to view specified role information. You can click **Reset** to reset the search criteria. Fuzzy search is supported.

Step 3 Click the target role instance to view its status and metric information.

Step 4 Customize and view monitoring graphs.

1. In the **Charts** area, click **Customize** to customize service monitoring metrics.
2. In **Period** area, select a time of period and click **View** to view the monitoring data within the time period.

----End

Managing Hosts

NOTE

For versions earlier than MRS 3.x, see **Managing Hosts**.

Step 1 On the MRS cluster details page, click the **Nodes** tab and expand a node group to view the host status.

The host list contains the **Node Name**, **IP Address**, **Rack**, **Operating Status**, **Health Status**, **CPU Usage**, **Memory Usage**, **Disk Usage**, **Network Speed**, Specification Name, **Specifications** and **AZ**.

- **Table 5-14** shows the host operating status.

Table 5-14 Host operating status

Status	Description
Normal	The host and service roles on the host are running properly.

Status	Description
Isolated	The host is isolated, and the service roles on the host stop running.

- Table 5-15 describes the host health status.

  Table 5-15 Host health status

Status	Description
Good	The host can properly send heartbeats.
Bad	The host fails to send heartbeats due to timeout.
Unknown	The host initial status is unknown during the operation of adding or deleting a host.

The nodes are sorted in ascending order by default. You can click **Node Name**, **IP Address**, **Rack**, **Operating Status**, **Health Status**, **CPU Usage**, **Memory Usage**, **Disk Usage**, **Network Speed**, **Specification Name**, or **Specifications** to change the sorting mode.

Step 2 Click the target node in the list to view its status and metric information.

----End

5.3 Cluster O&M

5.3.1 Importing and Exporting Data

Through the **Files** tab page, you can create, delete, import, export, delete files in the analysis cluster. Currently, file creation is not supported. Streaming clusters do not support the file management function on the MRS GUI. In a cluster with Kerberos authentication enabled, to read or write the folders in the root directory, add a role that has the required permissions on the folders by referring to **Creating a Role**. Then, add the new role to the user group to which the user who submits the job belongs by referring to **Related Tasks**.

Background

Data sources processed by MRS are from OBS or HDFS. OBS is an object-based storage service that provides you with massive, secure, reliable, and cost-effective data storage capabilities. MRS can process data in OBS directly. You can view, manage, and use data by using the web page of the management control platform or OBS client. In addition, you can use REST APIs independently or integrate APIs to service applications to manage and access data.

Before creating jobs, upload the local data to OBS for MRS to compute and analyze. MRS allows exporting data from OBS to HDFS for computing and analyzing. After the data analysis and computing are completed, you can store the

data in HDFS or export them to OBS. HDFS and OBS can also store the compressed data in the format of **bz2** or **gz**.

Importing Data

Currently, MRS can only import data from OBS to HDFS. The file upload rate decreases with the increase of the file size. This mode applies to scenarios where the data volume is small.

You can perform the following steps to import files and directories:

1. Log in to the MRS console.
2. Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's information.
3. Click the **Files** tab, and go to the file management page.
4. Select **HDFS File List**.
5. Go to the data storage directory, for example, **bd_app1**.

   The **bd_app1** directory is only an example. You can use any directory on the page or create a new one.

   The requirements for creating a folder are as follows:
   - The folder name contains a maximum of 255 characters
   - The folder name cannot be empty.
   - The folder name cannot contain the following special characters: /:*?"<>|\;&,'`!{}[]$%+
   - The value cannot start or end with a period (.).
   - The spaces at the beginning and end are ignored.
6. Click **Import Data** and configure the HDFS and OBS paths correctly. When configuring the OBS or HDFS path, click **Browse**, select a file directory, and click **Yes**.
   - OBS path
     - The path must start with **obs://**.
     - Files or programs encrypted by KMS cannot be imported.
     - An empty folder cannot be imported.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters: ;|&>,<'$*?\
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The OBS full path contains a maximum of 255 characters.
   - HDFS path
     - The path starts with **/user** by default.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters: ;|&>,<'$*?\

- The directory and file name cannot start or end with a space, but can contain spaces between them.
- The HDFS full path contains a maximum of 255 characters.

7. Click **OK**.

   You can view the file upload progress on the **File Operation Records** tab page. MRS processes the data import operation as a DistCp job. You can also check whether the DistCp job is successfully executed on the **Jobs** tab page.

Exporting Data

After the data analysis and computing are completed, you can store the data in HDFS or export them to OBS.

You can perform the following steps to export files and directories:

1. Log in to the MRS console.
2. Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.
3. Click the **Files** tab, and the file management page is displayed.
4. Select **HDFS File List**.
5. Go to the data storage directory, for example, **bd_app1**.
6. Click **Export Data** and configure the OBS and HDFS paths. When configuring the OBS or HDFS path, click **Browse**, select a file directory, and click **Yes**.
   - OBS path
     - The path must start with **obs://**.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters: ;|&>,<'$*?\
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The OBS full path contains a maximum of 255 characters.
   - HDFS path
     - The path starts with **/user** by default.
     - The directory and file name can contain letters, digits, hyphens (-), and underscores (_), but cannot contain the following special characters: ;|&>,<'$*?\
     - The directory and file name cannot start or end with a space, but can contain spaces between them.
     - The HDFS full path contains a maximum of 255 characters.

NOTE

When a folder is exported to OBS, a label file named **folder name_$folder$** is added to the OBS path. Ensure that the exported folder is not empty. If the exported folder is empty, OBS cannot display the folder and only generates a file named **folder name_$folder$**.

7. Click **OK**.

   You can view the file upload progress on the **File Operation Records** tab page. MRS processes the data export operation as a DistCp job. You can also check whether the DistCp job is successfully executed on the **Jobs** tab page.

Viewing Operation Logs

When importing and exporting data on the MRS management console, you can choose **Files** > **File Operation Records** to view the data import and export progress.

Table 5-16 describes the parameters of the file operation record.

Table 5-16 File operation record parameters

Parameter	Description
Submitted	Start time of data import or export.
Source Path	Source path of data. • OBS path during data import. • HDFS path during data export.
Target Path	Target path of data. • HDFS path during data import. • OBS path during data import.
Status	Status during data import or export. • Submitted • Accepted • Running • Completed • Terminated • Abnormal
Duration (min)	Time of data import or export. The unit is minute.
Result	Result of data import or export. • Successful • Failed • Killed • Undefined

Parameter	Description
Operation	View Log: allows you to view file operation logs.

5.3.2 Changing the Subnet of a Cluster

If the current subnet does not have sufficient IP addresses, you can change to another subnet in the same VPC of the current cluster to obtain more available subnet IP addresses. Changing a subnet does not affect the IP addresses or subnets of existing nodes.

For details about how to configure network ACL outbound rules, see **How Do I Configure a Network ACL Outbound Rule?**

Changing a Subnet When No Network ACL Is Associated

Step 1 Log in to the MRS console.

Step 2 Click the target cluster name to go to its details page.

Step 3 Click **Change Subnet** on the right of **Default Subnet**.

Step 4 Select the target subnet and click **OK**.

If no subnet is available, click **Create Subnet** to create a subnet first.

----End

Changing a Subnet When a Network ACL Is Associated

Step 1 Log in to the MRS console and click the target cluster to go to its details page.

Step 2 In the **Basic Information** area, view **VPC**.

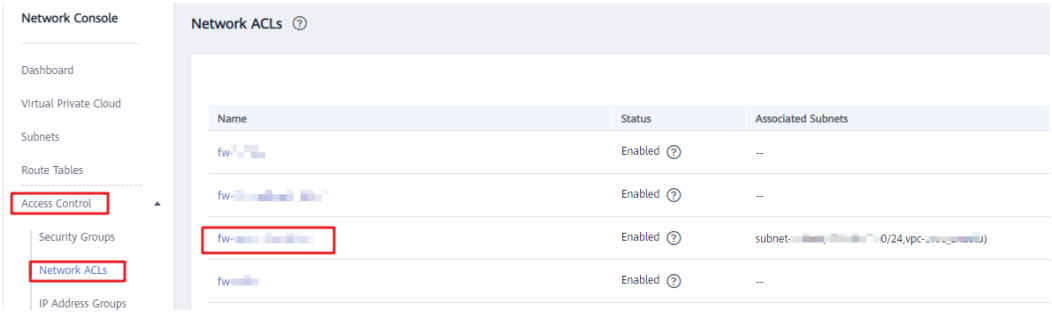
Step 3 Log in to the VPC console. In the navigation pane on the left, choose **Virtual Private Cloud** and obtain the IPv4 CIDR block corresponding to the VPC obtained in **Step 2**.

Step 4 Choose **Access Control** > **Network ACLs** and click the name of the network ACL that is associated with the default and new subnets.

NOTE

If both the default and new subnets are associated with a network ACL, add inbound rules to the network ACL by referring to **Step 5** to **Step 7**.

Figure 5-3 Network ACLs



Step 5 On the **Inbound Rules** page, choose **More** > **Insert Rule Above** in the **Operation** column.

Step 6 Add a network ACL rule. Set **Action** to **Allow**, **Source** to the VPC IPv4 CIDR block obtained in **Step 3**, and retain the default values for other parameters.

Step 7 Click **OK**.

NOTE

If you do not want to allow access from all IPv4 CIDR blocks of the VPC, add the IPv4 CIDR blocks of the default and new subnets by performing **Step 8** to **Step 12**. If the rules for VPC IPv4 CIDR blocks have been added, skip **Step 8** to **Step 12**.

Step 8 Log in to the MRS console.

Step 9 Click the target cluster to go to its details page.

Step 10 Click **Change Subnet** on the right of **Default Subnet**.

Step 11 Obtain the IPv4 CIDR blocks of the default and new subnets.

NOTICE

In this case, you do not need to click **OK** displayed in the **Change Subnet** dialog box. Otherwise, the default subnet will be updated to the new subnet, thereby making it difficult to query the IPv4 CIDR block of the default subnet. Exercise caution when performing this operation.

Step 12 Add the IPv4 CIDR blocks of the default and target subnets to the inbound rules of the network ACL bound to the two subnets by referring to **Step 4** to **Step 7**.

Step 13 Log in to the MRS console.

Step 14 Click the target cluster to go to its details page.

Step 15 Click **Change Subnet** on the right of **Default Subnet**.

Step 16 Select the target subnet and click **OK**.

----End

How Do I Configure a Network ACL Outbound Rule?

- Method 1

  Allow all outbound traffic. This method ensures that clusters can be created and used properly.

- Method 2

  Allow the mandatory outbound rules that can ensure the successful creation of clusters. You are not advised to use this method because created clusters may not run properly due to absent outbound rules. If the preceding problem occurs, contact O&M personnel.

  Similar to the example provided in method 1, set **Action** to **Allow** and add the outbound rules whose destinations are the address with **Secure Communications** enabled, NTP server address, OBS server address, OpenStack address, and DNS server address, respectively.

5.3.3 Configuring Message Notification

MRS uses SMN to offer a publish/subscribe model to achieve one-to-multiple message subscriptions and notifications in a variety of message types (SMSs and emails).

Scenario

On the MRS management console, you can enable or disable the notification service on the **Alarms** page. The functions in the following scenarios can be implemented only after the required cluster function is enabled:

- After a user subscribes to the notification service, the MRS management plane notifies the user of success or failure of manual cluster scale-out and scale-in, cluster termination, and auto scaling by emails or SMS messages.
- The management plane checks the alarms about the MRS cluster and sends a notification to the tenant if the alarms are critical.
- If either of the operations such as deletion, shutdown, specifications modification, restart, and OS update is performed on an ECS in a cluster, the MRS cluster works abnormally. The management plane notifies a user when detecting that the VM of the user is in either of the preceding operations.

Creating a Topic

A topic is a specified event for message publication and notification subscription. It serves as a message sending channel, where publishers and subscribers can interact with each other.

1. Log in to the management console.
2. Click **Service List**. Under **Management & Governance**, click **Simple Message Notification**.

   The **SMN** page is displayed.
3. In the navigation pane, choose **Topic Management** > **Topics**.

   The **Topics** page is displayed.
4. Click **Create Topic**.

   The **Create Topic** dialog box is displayed.
5. In **Topic Name**, enter a topic name. In **Display Name**, enter a display name.
6. Select an existing project from the **Enterprise Project** drop-down list, or click **Create Enterprise Project** to create an enterprise project on the **Enterprise Project Management** page and then select it.
7. Set tag keys and tag values. Tags consist of keys and values. They identify cloud resources so that you can easily categorize and search for your resources.

Adding Subscriptions to a Topic

To deliver messages published to a topic to subscribers, you must add subscription endpoints to the topic. SMN automatically sends a confirmation message to the subscription endpoint. The confirmation message is valid only within 48 hours. The

subscribers must confirm the subscription within 48 hours so that they can receive notification messages. Otherwise, the confirmation message becomes invalid, and you need to send it again.

1. Log in to the management console.
2. Under **Management & Governance**, click **Simple Message Notification**.

   The **SMN** page is displayed.
3. In the navigation pane, choose **Topic Management** > **Topics**.

   The **Topics** page is displayed.
4. Locate the topic to which you want to add a subscription, click **More** in the **Operation** column, and select **Add Subscription**.

   The **Add Subscription** box is displayed.

   Protocol can be set to **SMS**, FunctionGraph (function), **HTTP**, **HTTPS**, and **Email**.

   Endpoint indicates the address of the subscription endpoint. SMS and email, endpoints can be entered in batches. When adding endpoints in batches, each endpoint address occupies a line. You can enter a maximum of 10 endpoints.
5. Click **OK**.

The subscription you added is displayed in the subscription list.

Sending Notifications to Subscribers

1. Log in to the MRS console.
2. Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.
3. Click **Alarms**.
4. Choose **Notification Rules** > **Add Notification Rule**. The **Add Notification Rule** page is displayed.
5. Set the notification rule parameters.

Table 5-17 Parameters of a notification rule

Parameter	Description
Rule Name	User-defined notification rule name. Only digits, letters, hyphens (-), and underscores (_) are allowed.
Message Notification	• If you enable this function, the system sends notifications to subscribers based on the notification rule. • If you disable this function, the rule does not take effect, that is, notifications are not sent to subscribers.
Topic Name	Select an existing topic or click **Create Topic** to create a topic.

Parameter	Description
Notification Type	Select the type of the notification to be subscribed to. • Alarm
Subscription Items	Select the items to be subscribed to. You can select all or some items as required. Subscription rules in MRS 3.*x* or later: Alarm severity: critical, major, and minor Subscription rules in versions earlier than MRS 3.x: • Critical • Major • Minor • Suggestion

6. Click **OK**.

5.3.4 Checking Health Status

5.3.4.1 Before You Start

This section describes how to manage health checks on the MRS console.

Health check management operations on the MRS console apply only to clusters of **MRS 1.9.2 to MRS 2.1.x**.

Health check management on Manager applies to all versions. For MRS 3.x and later versions, see **Viewing a Health Check Task**. For versions earlier than MRS 3.x, see **Performing a Health Check**.

5.3.4.2 Performing a Health Check

Scenario

To ensure that cluster parameters, configurations, and monitoring are correct and that the cluster can run stably for a long time, you can perform a health check during routine maintenance.

NOTE

A system health check includes MRS Manager, service-level, and host-level health checks:

- MRS Manager health checks focus on whether the unified management platform can provide management functions.
- Service-level health checks focus on whether components can provide services properly.
- Host-level health checks focus on whether host indicators are normal.

The system health check includes three types of check items: health status, related alarms, and customized monitoring indicators for each check object. The health check results are not always the same as the **Health Status** on the portal.

Procedure

- Manually perform the health check for all services.

  On the MRS details page, choose **Management Operations** > **Start Cluster Health Check**.

  NOTE

  For the operations on MRS Manager, see **Performing a Health Check**; for the operations on FusionInsight Manager of MRS 3.*x* or later, see **Overview**.

  - The cluster health check includes Manager, service, and host status checks.
  - To perform cluster health checks, you can also choose **System** > **Check Health Status** > **Start Cluster Health Check** on MRS Manager.
  - To export the health check result, click **Export Report** in the upper left corner.
- Manually perform the health check for a service.
  a. On the MRS cluster details page, click **Components**.
  b. Select the target service from the service list.
  c. Choose **More** > **Start Service Health Check** to start the health check for the service.
- Manually perform the health check for a host.
  a. On the MRS details page, click **Nodes**.
  b. Expand the node group information and select the check box of the host to be checked.
  c. Choose **Node** > **Start Host Health Check** to start the health check for the host.

5.3.4.3 Viewing and Exporting a Health Check Report

Scenario

You can view the health check result on MRS and export it for further analysis.

NOTE

A system health check includes MRS Manager, service-level, and host-level health checks:

- MRS Manager health checks focus on whether the unified management platform can provide management functions.
- Service-level health checks focus on whether components can provide services properly.
- Host-level health checks focus on whether host indicators are normal.

The system health check includes three types of check items: health status, related alarms, and customized monitoring indicators for each check object. The health check results are not always the same as the **Health Status** on the portal.

Prerequisites

You have performed a health check.

Procedure

Step 1 On the MRS details page, choose **Management Operations** > **View Cluster Health Check Report**.

Step 2 Click **Export Report** on the health check report pane to export the report and view detailed information about check items.

----End

5.3.5 Remote O&M

5.3.5.1 Authorizing O&M

If you need technical support personnel to help you with troubleshooting, you can use the O&M authorization function to authorize technical support personnel to access your local host for fault location.

Procedure

Step 1 Log in to the MRS management console.

Step 2 In the navigation tree of the MRS management console, choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 In the upper right corner of the page, click **O&M**, choose **Authorize O&M**, and select the deadline for the support personnel to access the local host. Before the deadline, the support personnel have the temporary permission to access the local host.

Step 4 After the fault is rectified, click **O&M** in the upper right corner of the page and select **Cancel Authorization** to cancel the access permission for the support personnel.

----End

5.3.5.2 Sharing Logs

If you need technical support personnel to help you with troubleshooting, you can use the log sharing function to provide logs in a specific time to technical support personnel for fault location.

Procedure

Step 1 Log in to the MRS management console.

Step 2 In the navigation tree of the MRS management console, choose **Clusters** > **Active Clusters**, select a cluster, and click its name to switch to the cluster details page.

Step 3 In the upper right corner of the displayed page, choose **O&M** > **Share Log** to open the **Share Log** dialog box.

Step 4 Select the start time and end time in **Time Range**.

NOTE

- Select **Time Range** based on the suggestions of support personnel.
- **End Date** must be later than **Start Date**. Otherwise, logs cannot be filtered by time.

----End

5.3.6 Viewing MRS Operation Logs

You can view operation logs of clusters and jobs on the **Operation Logs** page. Log information is typically used for quickly locating faults in case of cluster exceptions, helping users resolve problems.

Operation Type

Currently, the following operation logs are provided by MRS. You can filter the logs in the search box.

- Cluster operations
  - Creating, deleting, scaling out, and scaling in a cluster
  - Creating and deleting a directory, deleting a file
- Job operations: Creating, stopping, and deleting a job
- Data operations: IAM user tasks, adding user, and adding user group

Log Fields

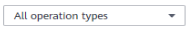
Logs are listed in chronological order by default in the log list, with the most recent logs displayed at the top.

Table 5-18 describes various fields in a log.

Table 5-18 Log description

Parameter	Description
Operation Type	Various types of operations, including: • Cluster operations • Job operations • Data operations
Operation IP	IP address where an operation is performed. **NOTE** If an MRS cluster fails to be deployed, the cluster is automatically deleted, and the operation logs of the automatically deleted cluster do not contain the **Operation IP** of the user.
Operation	Operation details. The value can contain a maximum of 2048 characters.
Time	Operation time. For a deleted cluster, only logs generated within the last six months are displayed. To view logs generated six months ago, contact technical support.
Enterprise Project	Enterprise project to which the cluster belongs

Table 5-19 Icon description

Icon	Description
	Select an enterprise project from the drop-down list box to filter logs.
	Select an operation type from the drop-down list box to filter logs. • **All Operation Types**: Filter all logs. • **Cluster**: Filter logs for **Cluster**. • **Job**: Filter logs for **Job**. • **Data**: Filter logs for **Data**.
	Filter logs by time. 1. Click the input box. 2. Specify the date and time. 3. Click **OK**. The left-side input box indicates the start time and the right-side one indicates the end time. The start time must be earlier than or equal to the end time. Otherwise, logs cannot be filtered.
	Enter a keyword of the **Operation Details** in the search box and click  to search for logs.

Icon	Description
	Click  to manually refresh the log list.

5.3.7 Terminating a Cluster

You can terminate an MRS cluster after job execution is complete.

Background

You can manually terminate a cluster after data analysis is complete or when the cluster encounters an exception. A cluster failed to be deployed will be automatically terminated.

Procedure

Step 1 Log in to the MRS console.

Step 2 In the navigation tree of the MRS console, choose **Clusters** > **Active Clusters**.

Step 3 Locate the cluster to be terminated, and click **Terminate** in the **Operation** column.

The cluster status changes from **Running** to **Terminating**, and finally to **Terminated**. You can view the clusters in **Terminated** state in **Cluster History**. The terminated cluster is no longer charged.

----End

5.4 Managing Nodes

5.4.1 Manually Scaling Out a Cluster

The storage and computing capabilities of MRS can be improved by simply adding Core nodes or Task nodes instead of modifying system architecture, reducing O&M costs. Core nodes can process and store data. You can add Core nodes to expand the node quantities and handle peak loads. Task nodes are used for computing and do not store persistent data.

Background

The MRS cluster supports a maximum of 500 Core and Task nodes. If more than 500 Core/Task nodes are required, contact technical support engineers or invoke a background interface to modify the database.

Core nodes and Task nodes can be added, excluding the Master node. Here, the maximum number of Core/Task nodes to be added is 500 minus the number of Core/Task nodes. For example, the current number of Core nodes is 3, the number of Core nodes to be added must be less than or equal to 497. If the cluster scale-out fails, you can add node to the cluster again.

If no node is added during cluster creation, you can specify the number of nodes to be added during scale-out. However, you cannot specify the nodes to be added.

The operations for scaling out a cluster vary depending on the selected version.

Procedure

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 Click the **Nodes** tab. In the **Operation** column of the node group, click **Scale Out**. The **Scale Out** page is displayed.

The scale-out operation can only be performed on the running clusters.

Step 4 Set **Scaled Out Nodes**, **Enable Component**, and **Run Bootstrap Action**, and click **OK**

NOTE

- If the Task node group does not exist in the cluster, configure the Task node by referring to **Adding a Task Node**.
- If a bootstrap action is added during cluster creation, the **Run Bootstrap Action** parameter is valid. If this function is enabled, the bootstrap actions added during cluster creation will be run on all the scaled out nodes.
- If the **New Specifications** parameter is available, the specifications that are the same as those of the original nodes have been sold out or discontinued. Nodes with new specifications will be added.
- Before scaling out the cluster, check whether its security group configuration is correct. Ensure that an inbound security group rule contains a rule in which **Protocol & Port** is set to **All**, and **Source** is set to a trusted accessible IP address range.

Step 5 In the **Scale Out Node** dialog box, click **OK**.

Step 6 A dialog box is displayed, indicating that the scale-out task is submitted successfully.

The following parameters explain the cluster scale-out process:

- Expanding: If a cluster is being expanded, its status is **Scaling out**. The submitted jobs will be executed and you can submit new jobs. You are not allowed to continue to scale out, or delete the cluster. You are advised not to restart the cluster or modify the cluster configuration.
- Expansion succeeded: If a cluster is expanded successfully, its status is **Running**.
- Failed scale-out: The cluster status is **Running** when the cluster scale-out failed. You can execute jobs and scale out the cluster again.

After the cluster is scaled out, you can view the node information of the cluster on the **Nodes** page.

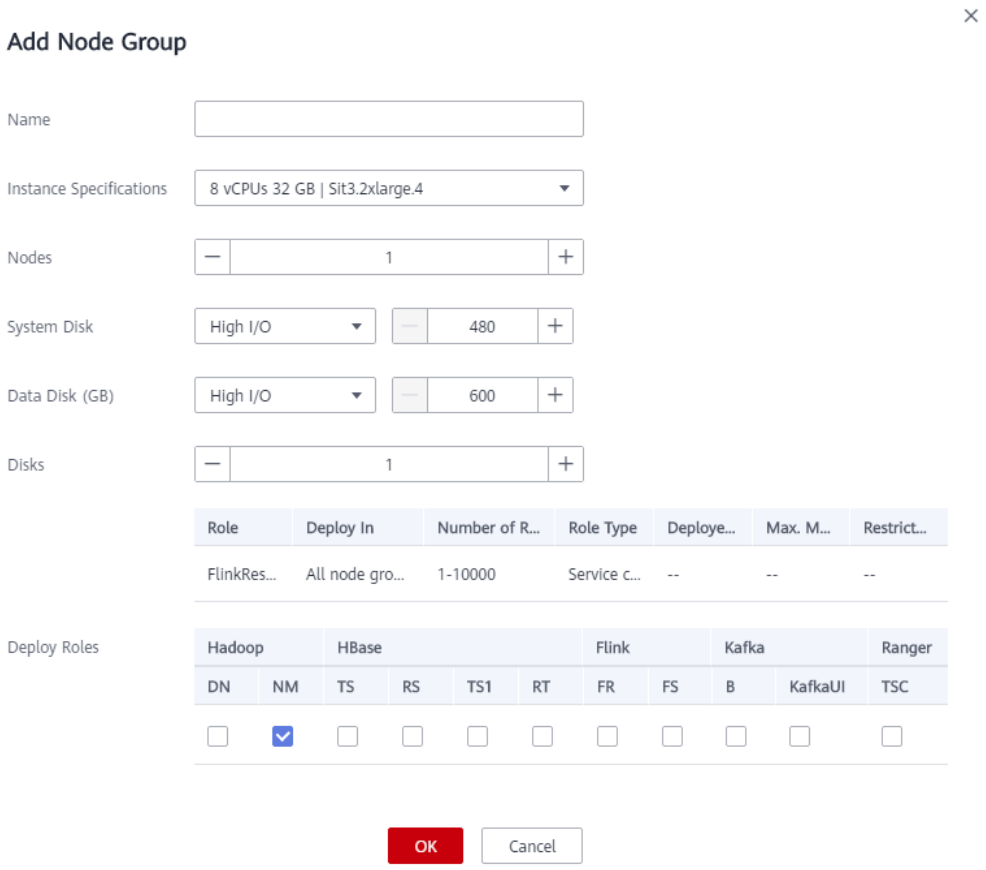
----End

Adding a Task Node

To add a task node to a custom cluster, perform the following steps:

1. On the cluster details page, click the **Nodes** tab and click **Add Node Group**. The **Add Node Group** page is displayed.
2. Select **NM** for **Deploy Roles** and set other parameters as required.

Figure 5-4 Adding a task node group



To add a task node to a non-custom cluster, perform the following steps:

1. On the cluster details page, click the **Nodes** tab and click **Configure Task Node**. The **Configure Task Node** page is displayed.
2. On the **Configure Task Node** page, set **Node Type**, **Instance Specifications**, **Nodes**, **System Disk**. In addition, if **Add Data Disk** is enabled, configure the storage type, size, and number of data disks.
3. Click **OK**.

Adding a Node Group

NOTE

Used to add node groups and applies to customized clusters of MRS 3.*x*.

1. On the cluster details page, click the **Nodes** tab and click **Add Node Group**. The **Add Node Group** page is displayed.
2. Set the parameters as needed.

Table 5-20 Parameters for adding a node group

Parameter	Description
Instance Specifications	Select the flavor type of the hosts in the node group.
Nodes	Set the number of nodes in the node group.
System Disk	Set the specifications and capacity of the system disk on the new node.
Data Disk (GB)	Set the specifications, capacity, and number of data disks of the new node.
Deploy Roles	Deploy the instances of each node in the new node group. The setting can be manually adjusted.

3. Click **OK**.

5.4.2 Manually Scaling In a Cluster

You can reduce the number of core or task nodes to scale in a cluster based on service requirements so that MRS delivers better storage and computing capabilities at lower O&M costs.

The scale-in operation is not allowed for a cluster that is performing active/standby synchronization.

Background

A cluster can have three types of nodes, master, core, and task nodes. Currently, only core and task nodes can be removed. To scale in a cluster, you only need to adjust the number of nodes on the MRS console. MRS then automatically selects the nodes to be removed.

The policies for MRS to automatically select nodes are as follows:

- MRS does not select the nodes with basic components installed, such as ZooKeeper, DBService, KrbServer, and LdapServer, because these basic components are the basis for the cluster to run.
- Core nodes store cluster service data. When scaling in a cluster, ensure that all data on the core nodes to be removed has been migrated to other nodes. You can perform follow-up scale-in operations only after all component services are decommissioned, for example, removing nodes from Manager and deleting ECSs. When selecting core nodes, MRS preferentially selects the nodes with a small amount of data and healthy instances to be decommissioned to prevent decommissioning failures. For example, if DataNodes are installed on core nodes in an analysis cluster, MRS preferentially selects the nodes with small data volume and good health status during scale-in.

  When core nodes are removed, their data is migrated to other nodes. If the user business has cached the data storage path, the client will automatically update the path, which may increase the service processing latency

temporarily. Cluster scale-in may slow the response of the first access to some HBase on HDFS data. You can restart HBase or disable or enable related tables to resolve this issue.

- Task nodes are computing nodes and do not store cluster data. Data migration is not involved in removing task nodes. Therefore, when selecting task nodes, MRS preferentially selects nodes whose health status is faulty, unknown, or subhealthy. On the **Components** tab of the MRS console, click a service and then the **Instances** tab to view the health status of the node instances.

Scale-In Verification Policy

To prevent component decommissioning failures, components provide different decommissioning constraints. Scale-in is allowed only when the constraints of all installed components are met. **Table 5-21** describes the scale-in verification policies.

Table 5-21 Decommissioning constraints

Component	Constraint
HDFS/DataNode	The number of available nodes after the scale-in is greater than or equal to the number of HDFS copies and the total HDFS data volume does not exceed 80% of the total HDFS cluster capacity. This ensures that the remaining space is sufficient for storing existing data after the scale-in and reserves some space for future use. **NOTE** To ensure data reliability, one backup is automatically generated for each file saved in HDFS, that is, two copies are generated in total.
HBase/ RegionServer	The total available memory of RegionServers on all nodes except the nodes to be removed is greater than 1.2 times of the memory which is currently used by RegionServers on these nodes. This ensures that the node to which the region on a decommissioned node is migrated has sufficient memory to bear the region of the decommissioned node.
Storm/ Supervisor	After the scale-in, ensure that the number of slots in the cluster is sufficient for running the submitted tasks. This prevents no sufficient resources being available for running the stream processing tasks after the scale-in.
Flume/ FlumeServer	If FlumeServer is installed on a node and Flume tasks have been configured for the node, the node cannot be deleted. This prevents the deployed service program from being deleted by mistake.

Scaling In a Cluster by Specifying the Node Quantity

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 Click the **Nodes** tab. In the **Operation** column of the node group, click **Scale In** to go to the **Scale In** page.

This operation can be performed only when the cluster and all nodes in it are running.

Step 4 Set **Scale-In Nodes** and click **OK**.

NOTE

- Before scaling in the cluster, check whether its security group configuration is correct. Ensure that an inbound security group rule contains a rule in which **Protocol & Port** is set to **All**, and **Source** is set to a trusted accessible IP address range.
- If damaged data blocks exist in HDFS, the cluster may fail to be scaled in. Contact technical support.

Step 5 A dialog box displayed in the upper right corner of the page indicates that the task of removing the node is submitted successfully.

The cluster scale-in process is explained as follows:

- During scale-in: The cluster status is **Scaling In**. The submitted jobs will be executed, and you can submit new jobs. You are not allowed to continue to scale in or terminate the cluster. You are advised not to restart the cluster or modify the cluster configuration.
- Successful scale-in: The cluster status is **Running**.
- Failed scale-in: The cluster status is **Running**. You can execute jobs or scale-in the cluster again.

After the cluster is scaled in, you can view the node information of the cluster on the **Nodes** page.

----End

5.4.3 Managing a Host (Node)

Scenario

To check an abnormal or faulty host (node), you need to stop all host roles on MRS. To recover host services after the host fault is rectified, restart all roles.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Nodes**.

Step 2 Unfold the node group information and select the check box of the target node.

Step 3 Choose **Node Operation** > **Start All Roles** or **Stop All Roles** to perform the required operation.

----End

5.4.4 Isolating a Host

Scenario

If a host is found to be abnormal or faulty, affecting cluster performance or preventing services from being provided, you can temporarily exclude that host from the available nodes in the cluster. In this way, the client can access other available nodes. In scenarios where patches are to be installed in a cluster, you can also exclude a specified node from patch installation.

You can isolate a host manually on MRS based on the actual service requirements or O&M plan. Only non-management nodes can be isolated.

Impact on the System

- After a host is isolated, all role instances on the host will be stopped. You cannot start, stop, or configure the host and any instances on the host.
- After a host is isolated, statistics of the monitoring status and indicator data of the host hardware and instances cannot be collected or displayed.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Nodes**.

Step 2 Unfold the node group information and select the check box of the target host.

Step 3 Choose **Node Operation** > **Isolate Host**.

Step 4 Confirm the information about the host to be isolated and click **OK**.

When **Operation successful** is displayed, click **Finish**. The host is isolated successfully, and the value of **Operating Status** becomes **Isolated**.

NOTE

For isolated hosts, you can cancel the isolation and add them to the cluster again. For details, see **Canceling Host Isolation**.

----End

5.4.5 Canceling Host Isolation

Scenario

After the exception or fault of a host is handled, you must cancel the isolation of the host for proper usage.

You can cancel the isolation of a host on MRS.

Prerequisites

- The host is in the **Isolated** state.
- The exception or fault of the host has been rectified.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Nodes**.

Step 2 Unfold the node group information and select the check box of the target host that you want to cancel its isolation.

Step 3 Choose **Node Operation** > **Cancel Host Isolation**.

Step 4 Confirm the information about the host for which the isolation is to be cancelled and click **OK**.

When **Operation successful** is displayed, click **Finish**. The host is de-isolated successfully, and the value of **Operating Status** becomes **Normal**.

Step 5 Select the host that has been de-isolated and choose **Node Operation** > **Start All Roles**.

----End

5.4.6 Scaling Up Master Node Specifications

As users' increasing services lead to Core node scale-out and high CPU usage, Master node specifications cannot meet user requirements and need to be scaled up. This section describes how to scale up Master node specifications.

Prerequisites

You have checked whether the Host Security Service (HSS) is enabled. If HSS is enabled, disable the HSS monitoring on the MRS cluster before upgrading the Master node specifications.

Use Restrictions

- Master nodes can be scaled up for clusters with two or more master nodes.
- The specifications of the Master node in a BMS cluster cannot be upgraded.

Master Node Specification Upgrade (One-Click Upgrade)

Step 1 Log in to the MRS console.

Step 2 In the left navigation pane, choose **Clusters** > **Active Clusters**, select the cluster for which you want to scale up Master node specifications, and click its name to switch to the cluster details page.

Step 3 On the **Nodes** tab page, select **Scale Up Specifications** in the **Operation** column of the Master node group. The **Scale Up Master Node Specifications** page is displayed.

Step 4 Select the target specifications and click **Submit Order**. The order has been submitted successfully.

The node specification scale-up takes some time. After the scale-up is successful, the cluster status changes to **Running**.

NOTE

- The VM to be scaled up is automatically stopped during the scale-up and started after the scale-up is complete.
- The scale-up does not automatically upgrade the memory of components due to different component usage requirements. You can adjust the memory of components as needed.

----End

5.5 Job Management

5.5.1 Introduction to MRS Jobs

An MRS job is the program execution platform of MRS. It is used to process and analyze user data. After a job is created, all job information is displayed on the **Jobs** tab page. You can view a list of all jobs and create and manage jobs. If the **Jobs** tab is not displayed on the cluster details page, submit a job in the background.

Data sources processed by MRS are from OBS or HDFS. OBS is an object-based storage service that provides you with massive, secure, reliable, and cost-effective data storage capabilities. MRS can process data in OBS directly. You can view, manage, and use data by using the web page of the management control platform or OBS client. In addition, you can use REST APIs independently or integrate APIs to service applications to manage and access data.

Before creating jobs, upload the local data to OBS for MRS to compute and analyze. MRS allows exporting data from OBS to HDFS for computing and analyzing. After the analyzing and computing are complete, you can store the data in HDFS or export them to OBS. HDFS and OBS can also store the compressed data in the format of **bz2** or **gz**.

Category

An MRS cluster allows creating and managing the following jobs: If a cluster in the **Running** state fails to create a job, check the health status of related

components on the cluster management page. For details, see **Viewing and Customizing Cluster Monitoring Metrics**.

- MapReduce: provides the capability of processing massive data quickly and in parallel. It is a distributed data processing mode and execution environment. MRS supports the submission of MapReduce JAR programs.
- Spark: a distributed in-memory computing framework. MRS supports SparkSubmit, Spark Script, and Spark SQL jobs.
  - SparkSubmit: You can submit the Spark JAR and Spark Python programs, execute the Spark Application, and compute and process user data.
  - SparkScript: You can submit the SparkScript scripts and batch execute Spark SQL statements.
  - Spark SQL: You can use Spark SQL statements (similar to SQL statements) to query and analyze user data in real time.
- Hive: an open-source data warehouse based on Hadoop. MRS allows you to submit HiveScript scripts and execute Hive SQL statements.
- Flink: provides a distributed big data processing engine that can perform stateful computations over both finite and infinite data streams.

Job List

Tasks are listed in chronological order by default in the task list, with the most recent jobs displayed at the top. **Table 5-22** describes the parameters in the job list.

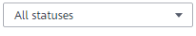
Table 5-22 Job list parameters

Parameter	Description
Name/ID	Job name, which is set when a job is created. ID is the unique identifier of a job. After a job is added, the system automatically assigns a value to ID.
Username	Name of the user who submits a job.

Parameter	Description
Type	The following data types are supported: • DistCp: importing and exporting data • MapReduce • Spark • SparkSubmit • SparkScript • Spark SQL • Hive SQL • HiveScript • Flink **NOTE** • After importing and exporting files on the **Files** tab page, you can view the DistCp job on the **Jobs** tab page. • Spark, Hive, and Flink jobs can be added only when the Spark, Hive, and Flink components are selected during cluster creation and the cluster is running.
Status	Job status. • Submitted • Accepted • Running • Completed • Terminated • Abnormal
Result	Execution result of a job. • Undefined: indicates that the job is being executed. • **Successful**: indicates that the job has been successfully executed. • **Killed**: indicates that the job is manually terminated during execution. • **Failed**: indicates that the job fails to be executed. **NOTE** Once a job has succeeded or failed, you cannot execute it again. However, you can add a job, and set job parameters to submit a job again.
Submitted	Time when a job is submitted.
Ended	Time when a job is completed or manually stopped.

Parameter	Description
Operation	• Viewing Log: Click **View Log** to view the real-time logs of running jobs. For details, see **Viewing Job Configuration and Logs**. • View Details: Click **View Details** to view the detailed configuration information about jobs. For details, see **Viewing Job Configuration and Logs**. • More – Stop: You can click **Stop** to stop a running job. For details, see **Stopping a Job**. – Delete: Click **Delete** to delete a job. For details, see **Deleting a Job**. – View Result: Click **View Result** to view the execution results of SparkSQL and SparkScript jobs whose status is **Completed** and result is **Successful**. **NOTE** • You cannot stop Spark SQL jobs. • A deleted job cannot be restored. Therefore, exercise caution when deleting a job. • If you choose to save job logs to OBS or HDFS, the system compresses and saves the logs to the corresponding path after the job execution is completed. Therefore, after a job execution of this type is completed, the job status is still **Running**. After the log is successfully stored, the job status changes to **Completed**. The log storage duration depends on the log size and takes several minutes.

Table 5-23 Icon description

Icon	Description
	Select a time range for job submission to filter jobs submitted in the time range.
	Select a certain job execution result from the drop-down list to display jobs of the status. • All statuses: Filter all jobs. • Successful: Filter jobs that are successfully executed. • Undefined: Filter jobs that are being executed. • Killed: Filter jobs that are manually stopped. • Failed: Filter jobs that fail to be executed.

Icon	Description
	Select a certain job type from the drop-down list to display jobs of the type. • All types • MapReduce • HiveScript • Distcp • SparkScript • Spark SQL • Hive SQL • SparkSubmit • Flink
	In the search box, search for a job by setting the corresponding search condition and click  . • Job name. • Job ID. • Username. • Queue name.
	Click  to manually refresh the job list.

Job Execution Permission Description

For a security cluster with Kerberos authentication enabled, a user needs to synchronize an IAM user before submitting a job on the MRS web UI. After the synchronization is completed, the MRS system generates a user with the same IAM username. Whether a user has the permission to submit jobs depends on the IAM policy bound to the user during IAM synchronization. For details about the job submission policy, see **Table 3-3** in **Synchronizing IAM Users to MRS**.

When a user submits a job that involves the resource usage of a specific component, such as accessing HDFS directories and Hive tables, user **admin** (Manager administrator) must grant the relevant permission to the user. Detailed operations are as follows:

Step 1 Log in to Manager as user **admin**.

Step 2 Add the role of the component whose permission is required by the user. For details, see **Creating a Role**.

Step 3 Change the user group to which the user who submits the job belongs and add the new component role to the user group. For details, see **Related Tasks**.

NOTE

After the component role bound to the user group to which the user belongs is modified, it takes some time for the role permissions to take effect.

----End

5.5.2 Running a MapReduce Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results. This section describes how to submit a MapReduce job on the MRS management console. MapReduce jobs are used to submit JAR programs to quickly process massive amounts of data in parallel and create a distributed data processing and execution environment.

If the job and file management functions are not supported on the cluster details page, submit the jobs in the background.

Prerequisites

You have uploaded the program packages and data files required for running jobs to OBS or HDFS.

Submitting a Job on the GUI

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. For details, see **Synchronizing IAM Users to MRS**.

NOTE

- When the policy of the user group to which the IAM user belongs changes from MRS ReadOnlyAccess to MRS CommonOperations, MRS FullAccess, or MRS Administrator, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from MRS CommonOperations, MRS FullAccess, or MRS Administrator to MRS ReadOnlyAccess, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

Step 4 Click the **Jobs** tab.

Step 5 Click **Create**. The **Create Job** page is displayed.

Step 6 In **Type**, select **MapReduce**. Configure other job information.

Table 5-24 Job configuration information

Parameter	Description
Name	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** You are advised to set different names for different jobs.
Program Path	Path of the program package to be executed. The following requirements must be met: • Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. • The path of the program to be executed can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **obs://**. Example: **obs://wordcount/program/*xxx*.jar** – HDFS: The path must start with **/user**. For details about how to import data to HDFS, see **Importing Data**. • For SparkScript and HiveScript, the path must end with **.sql**. For MapReduce, the path must end with **.jar**. For Flink and SparkSubmit, the path must end with **.jar** or **.py**. The **.sql**, **.jar**, and **.py** are case-insensitive.
Parameters	(Optional) It is the key parameter for program execution. Multiple parameters are separated by space. Configuration method: *Program class name Data input path Data output path* • Program class name: It is specified by a function in your program. MRS is responsible for transferring parameters only. • Data input path: Click **HDFS** or **OBS** to select a path or manually enter a correct path. • Data output path: Enter a directory that does not exist. The parameter contains a maximum of 150,000 characters. It cannot contain special characters ;|&><'$, but can be left blank. **CAUTION** If you enter a parameter with sensitive information (such as the login password), the parameter may be exposed in the job details display and log printing. Exercise caution when performing this operation.
Service Parameter	(Optional) It is used to modify service parameters for the job. The parameter modification applies only to the current job. To make the modification take effect permanently for the cluster, follow instructions in **Configuring Service Parameters**. To add multiple parameters, click ⊕ on the right. To delete a parameter, click **Delete** on the right. **Table 5-25** lists the common service configuration parameters.

Parameter	Description
Command Reference	Command submitted to the background for execution when a job is submitted.

Table 5-25 Service Parameter parameters

Parameter	Description	Example Value
fs.obs.access.key	Key ID for accessing OBS.	-
fs.obs.secret.key	Key corresponding to the key ID for accessing OBS.	-

Step 7 Confirm job configuration information and click **OK**.

After the job is created, you can manage it.

----End

Submitting a Job in the Background

In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 On the **Nodes** tab page, click the name of a Master node to go to the ECS management console.

Step 4 Click **Remote Login** in the upper right corner of the page.

Step 5 Enter the username and password of the Master node as prompted. The username is **root** and the password is the one set during cluster creation.

Step 6 Run the following command to initialize environment variables:

source /opt/Bigdata/client/bigdata_env

Step 7 If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If the Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 8 Run the following command to copy the program in the OBS file system to the Master node in the cluster:

hadoop fs -Dfs.obs.access.key=AK -Dfs.obs.secret.key=SK -copyToLocal source_path.jar target_path.jar

Example: **hadoop fs -Dfs.obs.access.key=XXXX -Dfs.obs.secret.key=XXXX -copyToLocal "obs://mrs-word/program/hadoop-mapreduce-examples-XXX.jar" "/home/omm/hadoop-mapreduce-examples-XXX.jar"**

You can log in to OBS Console using AK/SK. To obtain AK/SK information, click the username in the upper right corner of the management console and choose **My Credentials** > **Access Keys**.

Step 9 Run the following command to submit a wordcount job. If data needs to be read from OBS or outputted to OBS, the AK/SK parameters need to be added.

source /opt/Bigdata/client/bigdata_env;hadoop jar execute_jar wordcount input_path output_path

Example: **source /opt/Bigdata/client/bigdata_env;hadoop jar /home/omm/hadoop-mapreduce-examples-XXX.jar wordcount -Dfs.obs.access.key=XXXX -Dfs.obs.secret.key=XXXX "obs://mrs-word/input/*" "obs://mrs-word/output/"**

In the preceding command, **input_path** indicates a path for storing job input files on OBS. **output_path** indicates a path for storing job output files on OBS and needs to be set to a directory that does not exist

----End

5.5.3 Running a SparkSubmit Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results. This section describes how to submit a Spark job on the MRS console.

Prerequisites

You have uploaded the program packages and data files required for running jobs to OBS or HDFS.

Submitting a Job on the GUI

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. For details, see **Synchronizing IAM Users to MRS**.

NOTE

- When the policy of the user group to which the IAM user belongs changes from MRS ReadOnlyAccess to MRS CommonOperations, MRS FullAccess, or MRS Administrator, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from MRS CommonOperations, MRS FullAccess, or MRS Administrator to MRS ReadOnlyAccess, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

Step 4 Click the **Jobs** tab.

Step 5 Click **Create**. The **Create Job** page is displayed.

Step 6 Configure job information.

Table 5-26 Job configuration information

Parameter	Description
Name	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** You are advised to set different names for different jobs.
Program Path	Path of the program package to be executed. The following requirements must be met: • Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. • The path of the program to be executed can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **obs://**. Example: **obs://wordcount/program/*xxx*.jar** – HDFS: The path must start with **/user**. For details about how to import data to HDFS, see **Importing Data**. • For SparkScript and HiveScript, the path must end with **.sql**. For MapReduce, the path must end with **.jar**. For Flink and SparkSubmit, the path must end with **.jar** or **.py**. The **.sql**, **.jar**, and **.py** are case-insensitive.
Program Parameter	(Optional) Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **Table 5-27** describes the common parameters of a running program.

Parameter	Description
Parameters	(Optional) Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. Multiple parameters are separated by space. The parameter contains a maximum of 150,000 characters. It cannot contain special characters ;|&><'$, but can be left blank. **CAUTION** If you enter a parameter with sensitive information (such as the login password), the parameter may be exposed in the job details display and log printing. Exercise caution when performing this operation.
Service Parameter	(Optional) It is used to modify service parameters for the job. The parameter modification applies only to the current job. To make the modification take effect permanently for the cluster, follow instructions in **Configuring Service Parameters**. To add multiple parameters, click ⊕ on the right. To delete a parameter, click **Delete** on the right. **Table 5-28** lists the common service configuration parameters. **NOTE** If you need to run a long-term job, such as SparkStreaming, and access OBS, you need to use **Service Parameter** to import the AK/SK for accessing OBS.
Command Reference	Command submitted to the background for execution when a job is submitted.

Table 5-27 Program parameters

Parameter	Description	Example Value
--conf	Add the task configuration items.	spark.executor.memory=2G
--driver-memory	Set the running memory of driver.	2G
--num-executors	Set the number of executors to be started.	5
--executor-cores	Set the number of executor cores.	2
--class	Set the main class of a task.	org.apache.spark.examples.SparkPi
--files	Upload files to a task. The files can be custom configuration files or some data files from OBS or HDFS.	-
--jars	Upload additional dependency packages of a task to add the external dependency packages to the task.	-

Parameter	Description	Example Value
--executor-memory	Set executor memory.	2G
--conf spark-yarn.maxAppAttempts	Control the number of AM retries.	If this parameter is set to **0**, retry is not allowed. If this parameter is set to **1**, one retry is allowed.

Table 5-28 Service Parameter parameters

Parameter	Description	Example Value
fs.obs.access.key	Key ID for accessing OBS.	-
fs.obs.secret.key	Key corresponding to the key ID for accessing OBS.	-

Step 7 Confirm job configuration information and click **OK**.

After the job is created, you can manage it.

----End

Submitting a Job in the Background

In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.

Step 1 Create a user for submitting jobs. For details, see **Creating a User**.

In this example, a machine-machine user used in the user development scenario has been created, and user groups (**hadoop** and **supergroup**), the primary group (**supergroup**), and role permissions (**System_administrator** and **default**) have been correctly assigned to the user.

Step 2 Download the authentication credential.

- For clusters of MRS 3.*x* or later, log in to FusionInsight Manager and choose **System** > **Permission** > **User**. In the **Operation** column of the newly created user, choose **More** > **Download Authentication Credential**.
- For clusters whose version is earlier than MRS 3.*x*, log in to MRS Manager and choose **System** > **Manage User**. In the **Operation** column of the newly created user, choose **More** > **Download Authentication Credential**.

Step 3 Upload JAR files related to the job to the cluster. In this example, the sample JAR file built in Spark is used. It is stored in **$SPARK_HOME/examples/jars/**.

Step 4 Upload the authentication credential of the user created in **Step 2** to the **/opt/** directory of the cluster and run the following command to decompress the credential:

tar –xvf MRSTest *_xxxxxx_***keytab.tar**

You will obtain two files: **user.keytab** and **krb5.conf**.

Step 5 Before performing operations on the cluster, run the following commands:

source /opt/Bigdata/client/bigdata_env

cd $SPARK_HOME

Step 6 Run the following command to submit the Spark job:

./bin/spark-submit --master yarn --deploy-mode client --conf spark.yarn.principal=MRSTest --conf spark.yarn.keytab=/opt/user.keytab --class org.apache.spark.examples.SparkPi examples/jars/spark-examples_2.11-2.3.2-mrs-2.0.jar 10

Parameter description:

1. Computing capability of Yarn, which specifies that the job is submitted in client mode.
2. Configuration item of the Spark job. The authentication file and username are transferred here.
3. **spark.yarn.principal**: user created in step 1
4. **spark.yarn.keytab**: keytab file used for authentication
5. *xx***.jar**: JAR file used by the job

----End

5.5.4 Running a HiveSQL Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results. This section describes how to submit a HiveSQL job on the MRS management console. HiveSQL jobs are used to submit SQL statements and script files for data query and analysis. Both SQL statements and scripts are supported. If SQL statements contain sensitive information, use Script to submit them.

Prerequisites

You have uploaded the program packages and data files required for running jobs to OBS or HDFS.

Submitting a Job on the GUI

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. For details, see **Synchronizing IAM Users to MRS**.

NOTE

- When the policy of the user group to which the IAM user belongs changes from MRS ReadOnlyAccess to MRS CommonOperations, MRS FullAccess, or MRS Administrator, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from MRS CommonOperations, MRS FullAccess, or MRS Administrator to MRS ReadOnlyAccess, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

Step 4 Click the **Jobs** tab.

Step 5 Click **Create**. The **Create Job** page is displayed.

Step 6 Configure job information.

Table 5-29 Job configuration information

Parameter	Description
Name	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** You are advised to set different names for different jobs.
SQL Type	Submission type of the SQL statement • SQL • Script
SQL Statement	This parameter is valid only when **SQL Type** is set to **SQL**. Enter the SQL statement to be executed, and then click **Check** to check whether the SQL statement is correct. If you want to submit and execute multiple statements at the same time, use semicolons (;) to separate them.

Parameter	Description
SQL File	This parameter is valid only when **SQL Type** is set to **Script**. The path of the SQL file to be executed must meet the following requirements: • Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. • The path of the program to be executed can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **obs://**. Example: **obs://wordcount/program/*xxx*.jar** – HDFS: The path must start with **/user**. For details about how to import data to HDFS, see **Importing Data**. • For SparkScript and HiveScript, the path must end with **.sql**. For MapReduce, the path must end with **.jar**. For Flink and SparkSubmit, the path must end with **.jar** or **.py**. The **.sql**, **.jar**, and **.py** are case-insensitive. **NOTE** A file path on OBS can start with **obs://**. To submit jobs in this format, you need to configure permissions for accessing OBS. • If the OBS permission control function is enabled during cluster creation, you can use the **obs://** directory without extra configuration. • If the OBS permission control function is not enabled or is not supported when you create a cluster, configure the function by following instructions in **Accessing OBS**.
Program Parameter	(Optional) Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **Table 5-30** describes the common parameters of a running program.
Service Parameter	(Optional) It is used to modify service parameters for the job. The parameter modification applies only to the current job. To make the modification take effect permanently for the cluster, follow instructions in **Configuring Service Parameters**. To add multiple parameters, click ⊕ on the right. To delete a parameter, click **Delete** on the right. **Table 5-31** lists the common service configuration parameters.
Command Reference	Command submitted to the background for execution when a job is submitted.

Table 5-30 Program parameters

Parameter	Description	Example Value
--hiveconf	Hive service configuration, for example, set the execution engine to MapReduce.	Setting the execution engine to MR: **--hiveconf "hive.execution.engine=mr"**
--hivevar	Custom variable, for example, variable ID.	Setting the variable ID: **--hivevar id="123" select * from test where id = ${hivevar:id}**

Table 5-31 Service parameters

Parameter	Description	Example Value
fs.obs.access.key	Key ID for accessing OBS.	-
fs.obs.secret.key	Key corresponding to the key ID for accessing OBS.	-
hive.execution.engine	Engine for running a job.	• mr • tez

Step 7 Confirm job configuration information and click **OK**.

After the job is created, you can manage it.

----End

Submitting a Job in the Background

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 On the **Nodes** tab page, click the name of a Master node to go to the ECS management console.

Step 4 Click **Remote Login** in the upper right corner of the page.

Step 5 Enter the username and password of the Master node as prompted. The username is **root** and the password is the one set during cluster creation.

Step 6 Run the following command to initialize environment variables:

source /opt/BigData/client/bigdata_env

NOTE

- In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.
- If you use the client to connect to a specific Hive multi-instance in a scenario where multiple Hive instances are installed, run the following command to load the environment variables of the instance. Otherwise, skip this step. For example, load the environment variables of the Hive2 instance.

  source /opt/BigData/client/Hive2/component_env

Step 7 If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If the Kerberos authentication is disabled for the current cluster(normal mode), skip this step.

kinit *MRS cluster user* (The user must be in the **hive** user group.)

Step 8 Run the **beeline** command to connect to HiveServer and run tasks.

beeline

For clusters in normal mode, run the following commands. If no component service user is specified, the current OS user is used to log in to the HiveServer.

beeline -n *Component service user*

beeline -f *SQL files* (SQLs in the execution files)

----End

5.5.5 Running a SparkSql Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results. This section describes how to submit a SparkSQL job on the MRS console. SparkSQL jobs are used for data query and analysis. Both SQL statements and scripts are supported. If SQL statements contain sensitive information, use Spark Script to submit them.

Prerequisites

You have uploaded the program packages and data files required for running jobs to OBS or HDFS.

Submitting a Job on the GUI

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. For details, see **Synchronizing IAM Users to MRS**.

NOTE

- When the policy of the user group to which the IAM user belongs changes from MRS ReadOnlyAccess to MRS CommonOperations, MRS FullAccess, or MRS Administrator, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from MRS CommonOperations, MRS FullAccess, or MRS Administrator to MRS ReadOnlyAccess, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

Step 4 Click the **Jobs** tab.

Step 5 Click **Create**. On the displayed **Create Job** page, set **Type** to **SparkSql** and configure SparkSql job information by referring to **Table 5-32**.

Table 5-32 Job configuration information

Parameter	Description
Name	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** You are advised to set different names for different jobs.
SQL Type	Submission type of the SQL statement • SQL • Script
SQL Statement	This parameter is valid only when **SQL Type** is set to **SQL**. Enter the SQL statement to be executed, and then click **Check** to check whether the SQL statement is correct. If you want to submit and execute multiple statements at the same time, use semicolons (;) to separate them.

Parameter	Description
SQL File	This parameter is valid only when **SQL Type** is set to **Script**. The path of the SQL file to be executed must meet the following requirements: • Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. • The path of the program to be executed can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **obs://**. Example: **obs://wordcount/program/*xxx*.jar** – HDFS: The path must start with **/user**. For details about how to import data to HDFS, see **Importing Data**. • For SparkScript and HiveScript, the path must end with **.sql**. For MapReduce, the path must end with **.jar**. For Flink and SparkSubmit, the path must end with **.jar** or **.py**. The **.sql**, **.jar**, and **.py** are case-insensitive. **NOTE** A file path on OBS can start with **obs://**. To submit jobs in this format, you need to configure permissions for accessing OBS. • If the OBS permission control function is enabled during cluster creation, you can use the **obs://** directory without extra configuration. • If the OBS permission control function is not enabled or is not supported when you create a cluster, configure the function by following instructions in **Accessing OBS**.
Program Parameter	(Optional) Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **Table 5-33** describes the common parameters of a running program.
Service Parameter	(Optional) It is used to modify service parameters for the job. The parameter modification applies only to the current job. To make the modification take effect permanently for the cluster, follow instructions in **Configuring Service Parameters**. To add multiple parameters, click ⊕ on the right. To delete a parameter, click **Delete** on the right. **Table 5-34** lists the common service configuration parameters.
Command Reference	Command submitted to the background for execution when a job is submitted.

Table 5-33 Program parameters

Parameter	Description	Example Value
--conf	Task configuration items to be added.	spark.executor.memory=2G
--driver-memory	Running memory of a driver.	2G
--num-executors	Number of executors to be started.	5
--executor-cores	Number of executor cores.	2
--jars	Additional dependency packages of a task, which is used to add the external dependency packages to the task.	-
--executor-memory	Executor memory.	2G

Table 5-34 Service parameters

Parameter	Description	Example Value
fs.obs.access.key	Key ID for accessing OBS.	-
fs.obs.secret.key	Key corresponding to the key ID for accessing OBS.	-

Step 6 Confirm job configuration information and click **OK**.

After the job is created, you can manage it.

----End

Submitting a Job in the Background

In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.

Step 1 Create a user for submitting jobs. For details, see **Creating a User**.

In this example, a machine-machine user used in the user development scenario has been created, and user groups (**hadoop** and **supergroup**), the primary group (**supergroup**), and role permissions (**System_administrator** and **default**) have been correctly assigned to the user.

Step 2 Download the authentication credential.

- For clusters of MRS 3.*x* or later, log in to FusionInsight Manager and choose **System** > **Permission** > **User**. In the **Operation** column of the newly created user, choose **More** > **Download Authentication Credential**.

- For clusters whose version is earlier than MRS 3.*x*, log in to MRS Manager and choose **System** > **Manage User**. In the **Operation** column of the newly created user, choose **More** > **Download Authentication Credential**.

Step 3 Log in to the node where the Spark client is located, upload the user authentication credential created in 2 to the **/opt/** directory of the cluster, and run the following command to decompress the package:

tar –xvf MRSTest _xxxxxx_keytab.tar

After the decompression, you obtain the **user.keytab** and **krb5.conf** files.

Step 4 Before performing operations on the cluster, run the following commands:

source /opt/Bigdata/client/bigdata_env

cd $SPARK_HOME

Step 5 Open the **spark-sql** CLI and run the following SQL statement:

./bin/spark-sql --conf spark.yarn.principal=MRSTest --conf spark.yarn.keytab=/opt/user.keytab

To execute the SQL file, you need to upload the SQL file (for example, to the **/opt/** directory). After the file is uploaded, run the following command:

./bin/spark-sql --conf spark.yarn.principal=MRSTest --conf spark.yarn.keytab=/opt/user.keytab -f /opt/script.sql

----End

5.5.6 Running a Flink Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results. This section describes how to submit a Flink job on the MRS management console. Flink jobs are used to submit JAR programs to process streaming data.

Prerequisites

You have uploaded the program packages and data files required for running jobs to OBS or HDFS.

Submitting a Job on the GUI

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

In the **Basic Information** area on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. For details, see **Synchronizing IAM Users to MRS**.

NOTE

- When the policy of the user group to which the IAM user belongs changes from MRS ReadOnlyAccess to MRS CommonOperations, MRS FullAccess, or MRS Administrator, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from MRS CommonOperations, MRS FullAccess, or MRS Administrator to MRS ReadOnlyAccess, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

Step 4 Click the **Jobs** tab.

Step 5 Click **Create**. The **Create Job** page is displayed.

Step 6 Set **Type** to **Flink**. Configure Flink job information by referring to **Table 5-35**.

Table 5-35 Job configuration information

Parameter	Description
Name	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** You are advised to set different names for different jobs.
Program Path	Path of the program package to be executed. The following requirements must be met: • Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. • The path of the program to be executed can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **obs://**. Example: **obs://wordcount/program/*xxx*.jar** – HDFS: The path must start with **/user**. For details about how to import data to HDFS, see **Importing Data**.
Program Parameter	(Optional) Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **Table 5-36** describes the common parameters of a running program.

Parameter	Description
Parameters	(Optional) Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. Multiple parameters are separated by space. The parameter contains a maximum of 150,000 characters. It cannot contain special characters ;|&><'$, but can be left blank. **CAUTION** If you enter a parameter with sensitive information (such as the login password), the parameter may be exposed in the job details display and log printing. Exercise caution when performing this operation.
Service Parameter	(Optional) It is used to modify service parameters for the job. The parameter modification applies only to the current job. To make the modification take effect permanently for the cluster, follow instructions in **Configuring Service Parameters**. To add multiple parameters, click ⊕ on the right. To delete a parameter, click **Delete** on the right. **Table 5-37** describes the common parameters of a service.
Command Reference	Command submitted to the background for execution when a job is submitted.

Table 5-36 Program parameters

Parameter	Description	Example Value
-ytm	Memory size of each TaskManager container. (Optional unit. The unit is MB by default.)	1024
-yjm	Memory size of JobManager container. (Optional unit. The unit is MB by default.)	1024
-yn	Number of Yarn containers allocated to applications. The value is the same as the number of TaskManagers.	2
-ys	Number of TaskManager cores.	2
-ynm	Custom name of an application on Yarn.	test
-c	Class of the program entry point (for example, the **main** or **getPlan()** method). This parameter is required only when the JAR file does not specify the class of its manifest.	com.bigdata.mrs.test

NOTE

For MRS 3.x or later, the **-yn** parameter is not supported.

Table 5-37 Service parameters

Parameter	Description	Example Value
fs.obs.access.key	Key ID for accessing OBS.	-
fs.obs.secret.key	Key corresponding to the key ID for accessing OBS.	-

Step 7 Confirm job configuration information and click **OK**.

After the job is created, you can manage it.

----End

Submitting a Job in the Background

In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.

Step 1 Log in to the MRS client.

Step 2 Run the following command to initialize environment variables:

source /opt/Bigdata/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the cluster, perform the following steps. If Kerberos authentication is not enabled for the cluster, skip this step.

1. Prepare a user for submitting Flink jobs.
2. Log in to Manager as the newly created user.
   - For MRS 3.x earlier: Log in to Manager of the cluster. Choose **System** > **Manage User**. In the **Operation** column of the row that contains the added user, choose **More** > **Download authentication credential** to locate the row that contains the user.
   - For MRS 3.*x* or later: Log in to Manager of the cluster. Choose **System** > **Permission** > **Manage User**. On the displayed page, locate the row that contains the added user, click **More** in the **Operation** column, and select **Download authentication credential**.
3. Decompress the downloaded authentication credential package and copy the **user.keytab** file to the client node, for example, to the **/opt/Bigdata/client/Flink/flink/conf** directory on the client node. If the client is installed on a node outside the cluster, copy the **krb5.conf** file to the **/etc/** directory on this node.
4. For MRS 3.x or later: In security mode, add the service IP address of the node where the client is installed and floating IP address of Manager to the **jobmanager.web.allow-access-address** configuration item in the **/opt/Bigdata/client/Flink/flink/conf/flink-conf.yaml** file.

5. Run the following commands to configure security authentication by adding the **keytab** path and username to the **/opt/Bigdata/client/Flink/flink/conf/flink-conf.yaml** configuration file.

   security.kerberos.login.keytab: *<user.keytab file path>*

   security.kerberos.login.principal: *<Username>*

   Example:

   security.kerberos.login.keytab: /opt/Bigdata/client/Flink/flink/conf/user.keytab

   security.kerberos.login.principal: test

6. Run the following command to perform security hardening in the **bin** directory of the Flink client. Set password to a new password for submitting jobs.

   sh generate_keystore.sh <*password*>

   This script automatically replaces the SSL value in the **/opt/Bigdata/client/Flink/flink/conf/flink-conf.yaml** file. For MRS 3.x or earlier, external SSL is disabled by default in security clusters. To enable external SSL, run this script again after configuration. The configuration parameters do not exist in the default Flink configuration of MRS, if you enable SSL for external connections, you need to add the parameters listed in **Table 5-38**.

Table 5-38 Parameter description

Parameter	Example Value	Description
security.ssl.rest.enabled	true	Switch to enable external SSL.
security.ssl.rest.keystore	${path}/flink.keystore	Path for storing **keystore**.
security.ssl.rest.keystore-password	123456	Password of the **keystore**. **123456** indicates a user-defined password is required.
security.ssl.rest.key-password	123456	Password of the SSL key. **123456** indicates a user-defined password is required.
security.ssl.rest.truststore	${path}/flink.truststore	Path for storing the **truststore**.
security.ssl.rest.truststore-password	123456	Password of the **truststore**. **123456** indicates a user-defined password is required.

NOTE

- For MRS 3.x or earlier: The **generate_keystore.sh** script is automatically generated.
- Perform **Authentication and Encryption**. The generated **flink.keystore**, **flink.truststore**, and **security.cookie** files are automatically filled in the corresponding configuration items in **flink-conf.yaml**.
- For MRS 3.*x* or later: You can obtain the values of **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password** using the Manager plaintext encryption API by running the following command:

  curl -k -i -u <*user name*>:<*password*> -X POST -HContent-type:application/json -d '{"plainText":"<*password*>"}' 'https://*x.x.x.x*:28443/web/api/v2/tools/encrypt'; In the preceding command, <*password*> must be the same as the password used for issuing the certificate, and *x.x.x.x* indicates the floating IP address of Manager in the cluster.

7. Configure paths for the client to access the **flink.keystore** and **flink.truststore** files.
   - Absolute path: After the script is executed, the file path of **flink.keystore** and **flink.truststore** is automatically set to the absolute path **opt/Bigdata/client/Flink/flink/conf/** in the **flink-conf.yaml** file. In this case, you need to move the **flink.keystore** and **flink.truststore** files from the **conf** directory to this absolute path on the Flink client and Yarn nodes.
   - Relative path: Perform the following steps to set the file path of **flink.keystore** and **flink.truststore** to the relative path and ensure that the directory where the Flink client command is executed can directly access the relative paths.
     i. In the **/opt/Bigdata/client/Flink/flink/conf/**directory, create a new directory, for example, **ssl**.
     ii. Move the **flink.keystore** and **flink.truststore** file to the /**opt/Bigdata/client/Flink/flink/conf/ssl/** directory.
     iii. For MRS 3.*x* or later: Change the values of the following parameters in the **flink-conf.yaml** file to relative paths:
     ```
     security.ssl.keystore: ssl/flink.keystore
     security.ssl.truststore: ssl/flink.truststore
     ```
     iv. For MRS 3.x or earlier: Change the values of the following parameters in the **flink-conf.yaml** file to relative paths:
     ```
     security.ssl.internal.keystore: ssl/flink.keystore
     security.ssl.internal.truststore: ssl/flink.truststore
     ```
8. If the client is installed on a node outside the cluster, add the following configuration to the configuration file (for example, **/opt/Bigdata/client/Flink/fink/conf/flink-conf.yaml**). Replace ***xx.xx.xxx.xxx*** with the IP address of the node where the client resides.
   ```
   web.access-control-allow-origin: xx.xx.xxx.xxx
   jobmanager.web.allow-access-address: xx.xx.xxx.xxx
   ```

Step 4 Run a wordcount job.

- Normal cluster (Kerberos authentication disabled)
  - Run the following commands to start a session and submit a job in the session:
    ```
    yarn-session.sh -nm "session-name"
    flink run /opt/Bigdata/client/Flink/flink/examples/streaming/WordCount.jar
    ```
  - Run the following command to submit a single job on Yarn:
    ```
    flink run -m yarn-cluster /opt/Bigdata/client/Flink/flink/examples/streaming/WordCount.jar
    ```

- Security cluster (Kerberos authentication enabled)
  - If the **flink.keystore** and **flink.truststore** file are stored in the absolute path:
    - Run the following commands to start a session and submit a job in the session:
      ```
      yarn-session.sh -nm "session-name"
      flink run /opt/Bigdata/client/Flink/flink/examples/streaming/WordCount.jar
      ```
    - Run the following command to submit a single job on Yarn:
      ```
      flink run -m yarn-cluster /opt/Bigdata/client/Flink/flink/examples/streaming/WordCount.jar
      ```
  - If the **flink.keystore** and **flink.truststore** file are stored in the relative path:
    - In the same directory of SSL, run the following command to start a session and submit jobs in the session. The SSL directory is a relative path. For example, if the SSL directory is **opt/Bigdata/client/Flink/flink/conf/**, then run the following command in this directory:
      ```
      yarn-session.sh -t ssl/ -nm "session-name"
      flink run /opt/Bigdata/client/Flink/flink/examples/streaming/WordCount.jar
      ```
    - Run the following command to submit a single job on Yarn:
      ```
      flink run -m yarn-cluster -yt ssl/ /opt/Bigdata/client/Flink/flink/examples/streaming/WordCount.jar
      ```

----End

5.5.7 Running a Kafka Job

You can submit programs developed by yourself to MRS to execute them, and obtain the results. This topic describes how to generate and consume messages in a Kafka topic.

Currently, Kafka jobs cannot be submitted on the GUI. You can submit them in the background.

Submitting a Job in the Background

Query the instance addresses of ZooKeeper and Kafka, and then run the Kafka job.

Querying the Instance Address (3.x)

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 Go to the FusionInsight Manager page. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. On MRS Manager, choose **Services** > **ZooKeeper** > **Instance** to query the IP addresses of ZooKeeper instances. Record any IP address of a ZooKeeper instance.

Step 4 Choose **Services** > **Kafka** > **Instance** to query the IP addresses of Kafka instances. Record any IP address of a Kafka instance.

----End

Querying the Instance Address (Versions Earlier Than 3.x)

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 On the MRS cluster details page, choose **Components > ZooKeeper > Instance** to query the IP addresses of ZooKeeper instances. Record any IP address of a ZooKeeper instance.

Step 4 Choose **Components > Kafka > Instance** to query the IP addresses of Kafka instances. Record any IP address of a Kafka instance.

----End

Running a Kafka Job

In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.

Step 1 On the **Nodes** tab page of the cluster detail page, click the name of the Master2 node to go to the ECS management console.

Step 2 Click **Remote Login** in the upper right corner of the page.

Step 3 Enter the username and password of the Master node as prompted. The username is **root** and the password is the one set during cluster creation.

Step 4 Run the following command to initialize environment variables:

source /opt/Bigdata/client/bigdata_env

Step 5 If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If the Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 6 Run the following command to create a Kafka topic:

kafka-topics.sh --create --zookeeper <IP address of the ZooKeeper role instance:2181/kafka> --partitions 2 --replication-factor 2 --topic <Topic name>

Step 7 Produce messages in a topic test.

Run the following command: **kafka-console-producer.sh --broker-list <IP address of the Kafka role instance:9092> --topic <Topic name> --producer.config /opt/Bigdata/client/Kafka/kafka/config/producer.properties**.

Input specified information as the messages produced by the producer and then press **Enter** to send the messages. To end message production, press **Ctrl+C** to exit.

Step 8 Consume messages in the topic test.

kafka-console-consumer.sh --topic <Topic name> --bootstrap-server <Kafka role instance IP:210079092> --consumer.config /opt/Bigdata/client/Kafka/kafka/config/consumer.properties

NOTE

If Kerberos authentication is enabled in the cluster, change the port number 9092 to 21007 when running the preceding two commands. For details, see **List of Open Source Component Ports**.

----End

5.5.8 Viewing Job Configuration and Logs

This section describes how to view job configuration and logs.

Background

- You can view configuration information of all jobs.
- You can only view logs of running jobs.

  Because logs of Spark SQL and DistCp jobs are not in the background, you cannot view logs of running Spark SQL and DistCp jobs.

Procedure

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 Click **Jobs**.

Step 4 In the **Operation** column of the job to be viewed, click **View Details**.

In the **View Details** window that is displayed, configuration of the selected job is displayed.

Step 5 Select a running job, and click **View Log** in the **Operation** column.

In the new page that is displayed, real-time log information of the job is displayed.

Each tenant can submit and view 10 jobs concurrently.

----End

5.5.9 Stopping a Job

This section describes how to stop running MRS jobs.

Background

You cannot stop Spark SQL jobs. After a job is stopped, its status changes to **Terminated** and the job cannot be executed again.

Procedure

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name.

The cluster details page is displayed.

Step 3 Click **Jobs**.

Step 4 Select a running job, and choose **More > Stop** in the **Operation** column.

The job status changes from **Running** to **Terminated**.

----End

5.5.10 Deleting a Job

This section describes how to delete an MRS job. After a job is executed, you can delete it if you do not need to view its information.

Background

Jobs can be deleted one after another or in a batch. A deleted job cannot be restored. Therefore, exercise caution when deleting a job.

Procedure

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name.

The cluster details page is displayed.

Step 3 Click **Jobs**.

Step 4 Choose **More > Delete** from the **Operation** in the row of the target job to be deleted.

In this step, you can only delete one job only.

Step 5 If you select multiple jobs and click **Delete** on the upper left of the job list.

You can delete one, multiple, or all jobs.

----End

5.5.11 Using Encrypted OBS Data for Job Running

In encrypted data in OBS file systems can be used to run jobs, and the encrypted job running results can be stored in OBS file systems. Currently, data can be accessed only through an OBS protocol.

OBS supports data encryption and decryption using KMS keys. All encryption and decryption operations are performed on OBS, and keys are managed by DEW.

To use the OBS encryption function in MRS, you must have the KMS Administrator permissions and configure the following settings for the corresponding component:

NOTE

If the **OBS permission control** function is enabled in a cluster, the default agency **MRS_ECS_DEFAULT_AGENCY** configured on the ECS or the AK/SK of the custom agency is used for accessing OBS. OBS uses the received AK/SK to access DEW to obtain the KMS key status. Therefore, you need to bind the KMS Administrator policy to the used agency. Otherwise, OBS returns the "403 Forbidden" error when processing encrypted data. Currently, the KMS Administrator policy is bound to the agency **MRS_ECS_DEFAULT_AGENCY** by default. If you use a custom agency, you need to manually bind the policy to your custom agency.

Prerequisites

You have configured the function of accessing OBS from MRS first to use the OBS encryption function. For details, see **Configuring a Storage-Compute Decoupled Cluster (Agency)**.

Hive Configuration

Step 1 Log in to the MRS management console. In the navigation tree on the left, choose **Clusters** > **Active Clusters** and click the cluster name.

Step 2 Choose **Components** > **Hive** > **Service Configuration**.

Step 3 Switch **Basic** to **All**, and search for and set the following parameters:

Table 5-39 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	(Optional) This parameter indicates an ID of the KMS key used for encryption. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK**.

----End

Hadoop Configuration

Method 1: Configuration on the GUI

Step 1 Log in to the MRS management console. In the navigation tree on the left, choose **Clusters** > **Active Clusters** and click the cluster name.

Step 2 Choose **Components** > **HDFS** > **Service Configuration**.

Step 3 Switch **Basic** to **All**, and search for and set the following parameters:

Table 5-40 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK**.

Step 5 Log in to the Master node as user **root**. The password is the password of user **root** you set when you create the cluster. If the cluster has multiple Master nodes, log in to each Master node and repeat **Step 5** to **Step 7**.

Step 6 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 7 Run the following command to update client configurations, and enter the username and password. The username is **admin**, and the password is the password of user **admin** you set when you create the cluster.

./ autoRefreshConfig.sh

----End

Method 2: Configuration Through the Client Configuration File

Add the following parameter settings to the client configuration file, for example, **/opt/Bigdata/client/HDFS/hadoop/etc/hadoop/core-site.xml**, on the Master node. If the cluster has multiple Master nodes, log in to each Master node and perform this operation.

Table 5-41 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

HBase Configuration

Method 1: Configuration on the GUI

Step 1 Log in to the MRS management console. In the navigation tree on the left, choose **Clusters** > **Active Clusters** and click the cluster name.

Step 2 Choose **Components** > **HBase** > **Service Configuration**.

Step 3 Switch **Basic** to **All**, and search for and set the following parameters:

Table 5-42 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK**.

Step 5 Log in to the Master node as user **root**. The password is the password of user **root** you set when you create the cluster. If the cluster has multiple Master nodes, log in to each Master node and repeat **Step 5** to **Step 7**.

Step 6 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 7 Run the following command to update client configurations, and enter the username and password. The username is **admin**, and the password is the password of user **admin** you set when you create the cluster.

./ autoRefreshConfig.sh

----End

Method 2: Configuration Through the Client Configuration File

Add the following parameter settings to the client configuration file, for example, **/opt/Bigdata/client/HBase/hbase/conf/core-site.xml**, on the Master node. If the cluster has multiple Master nodes, log in to each Master node and perform this operation.

Table 5-43 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Spark Configuration

Method 1: Configuration on the GUI

Step 1 Log in to the MRS management console. In the navigation tree on the left, choose **Clusters** > **Active Clusters** and click the cluster name.

Step 2 Choose **Components** > **Spark** > **Service Configuration**.

Step 3 Switch **Basic** to **All**, and search for and set the following parameters:

Table 5-44 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.

Parameter	Value	Description
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK**.

Step 5 Log in to the Master node as user **root**. The password is the password of user **root** you set when you create the cluster. If the cluster has multiple Master nodes, log in to each Master node and repeat **Step 5** to **Step 7**.

Step 6 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 7 Run the following command to update client configurations, and enter the username and password. The username is **admin**, and the password is the password of user **admin** you set when you create the cluster.

./autoRefreshConfig.sh

----End

Method 2: Configuration Through the Client Configuration File

Add the following parameter settings to the client configuration file, for example, **/opt/Bigdata/client/Spark/spark/conf/core-site.xml**, on the Master node. If the cluster has multiple Master nodes, log in to each Master node and perform this operation.

Table 5-45 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.

Parameter	Value	Description
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Presto Configuration

Step 1 Log in to the MRS management console. In the navigation tree on the left, choose **Clusters** > **Active Clusters** and click the cluster name.

Step 2 Choose **Components** > **Presto** > **Service Configuration**.

Step 3 Switch **Basic** to **All**, and search for and set the following parameters:

Table 5-46 Data encryption parameters

Parameter	Value	Description
fs.obs.server-side-encryption-type	SSE-KMS	• **SSE-KMS**: KMS keys are used for encryption and decryption • **NONE**: The encryption function is disabled.
fs.obs.server-side-encryption-key	-	ID of the KMS key used for encryption. This parameter is optional. If **fs.obs.server-side-encryption-type** is set to **SSE-KMS** and this parameter is not set, OBS uses the default KMS key for encryption.
fs.obs.connection.ssl.enabled	true	Whether to establish a secure connection with OBS. • **true**: The secure connection is enabled. To use OBS encryption and decryption, this parameter must be set to **true**. • **false**: The secure connection is disabled.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK**.

----End

5.5.12 Configuring Job Notification Rules

MRS uses SMN to offer a publish/subscribe model to achieve one-to-multiple message subscriptions and notifications in a variety of message types (SMSs and emails). You can configure job notification rules to receive notifications immediately upon a job execution success or failure.

Procedure

Step 1 Log in to the management console.

Step 2 Click **Service List**. Under **Management & Governance**, click **Simple Message Notification**.

Step 3 Create a topic and add subscriptions to the topic. For details, see **Configuring Message Notification**.

Step 4 Go to the MRS management console, and click the cluster name to go to the cluster details page.

Step 5 Click the **Alarms** tab, and choose **Notification Rules** > **Add Notification Rule**.

Step 6 Configure a notification rule for sending job execution results to subscribers.

Table 5-47 Parameters of adding a notification rule

Parameter	Description
Rule Name	User-defined notification rule name. Only digits, letters, hyphens (-), and underscores (_) are allowed.
Message Notification	If you enable this function, subscription messages will be sent to subscribers.
Topic Name	Select an existing topic or click **Create Topic** to create a topic.
Notification Type	Select **Event**.
Subscription Items	1. Click ⊞ next to **Suggestion**. 2. Click ⊞ next to **Manager**. 3. Select **Job Running Succeeded** and **Job Running Failed**.

----End

5.6 Component Management

5.6.1 Object Management

MRS contains different types of basic objects. **Table 5-48** describes these objects.

Table 5-48 MRS basic object overview

Object	Description	Example
Service	Function set that can complete specific business.	KrbServer service and LdapServer service
Service instance	Specific instance of a service, usually called service.	KrbServer service
Service role	Function entity that forms a complete service, usually called role.	KrbServer is composed of the KerberosAdmin role and KerberosServer role.
Role instance	Specific instance of a service role running on a host.	KerberosAdmin that is running on Host2 and KerberosServer that is running on Host3
Host	An ECS running Linux OS.	Host1 to Host5
Rack	Physical entity that contains multiple hosts connecting to the same switch.	Rack1 contains Host1 to Host5.
Cluster	Logical entity that consists of multiple hosts and provides various services.	Cluster1 cluster consists of five hosts (Host1 to Host5) and provides services such as KrbServer and LdapServer.

5.6.2 Viewing Configuration

On MRS, you can view the configuration of services (including roles) and role instances.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

- Query service configuration.
  a. On the MRS cluster details page, click **Components**.
  b. Select the target service from the service list.
  c. Click **Service Configuration**.
  d. Switch **Basic** to **All**. All configuration parameters of the service are displayed in the navigation tree. The service name and role names are displayed from upper to lower in the navigation tree.
  e. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

The parameters under the service nodes and role nodes are service configuration parameters and role configuration parameters respectively.

f. Select **Non-default** from the **--Select--** drop-down list. The parameters whose values are not default values are displayed.

- Query role instance configurations.

a. On the MRS cluster details page, click **Components**.

b. Select the target service from the service list.

c. Click the **Instances** tab.

d. Click the target role instance from the role instance list.

e. Click **Instance Configuration**.

f. Switch **Basic** to **All** on the right of the page. All configuration parameters of the role instance are displayed in the navigation tree.

g. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

h. Select **Non-default** from the **--Select--** drop-down list. The parameters whose values are not default values are displayed.

5.6.3 Managing Services

You can perform the following operations on MRS:

- Start the service in the **Stopped**, **Stop Failed**, or **Failed to Start** state to use the service.
- Stop the services or stop abnormal services.
- Restart abnormal services or configure expired services to restore or enable the services.

Prerequisites

- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Impact on the System

- The stateful component cannot be added to the task node group.

Starting, Stopping, and Restarting a Service

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Locate the row that contains the target service, **Start**, **Stop**, and **Restart** to start, stop, or restart the service.

Services are interrelated. If a service is started, stopped, and restarted, services dependent on it will be affected.

The services will be affected in the following ways:

- If a service is to be started, the lower-layer services dependent on it must be started first.

- If a service is stopped, the upper-layer services dependent on it are unavailable.
- If a service is restarted, the running upper-layer services dependent on it must be restarted.

----End

5.6.4 Configuring Service Parameters

On the MRS console, you can view and modify the default service configurations based on site requirements and export or import the configurations.

Impact on the System

- You need to download and update the client configuration files after configuring HBase, HDFS, Hive, Spark, Yarn, and MapReduce service properties.
- The parameters of DBService cannot be modified when only one DBService role instance exists in the cluster.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Modifying Service Parameters

1. On the MRS cluster details page, click **Components**.
2. Select the target service from the service list.
3. Click **Service Configuration**.
4. Switch **Basic** to **All**. All configuration parameters of the service are displayed in the navigation tree. The service name and role names are displayed from upper to lower in the navigation tree.
5. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

   If you want to cancel the modification of a parameter value, click to restore it.
6. Click **Save Configuration**, select **Restart the affected services or instances**, and click **OK**.

   NOTE

   To update the queue configuration of Yarn without restarting service, choose **More** > **Refresh Queue** on the **Service Status** tab page to update the queue for the configuration to take effect.

5.6.5 Configuring Customized Service Parameters

Each component of MRS supports all open-source parameters. MRS supports the modification of some parameters for key application scenarios. Some component clients may not include all parameters with open-source features. To modify the

component parameters that are not directly supported by MRS, you can add new parameters for components by using the configuration customization function on MRS. Newly added parameters are saved in component configuration files and take effect after restart.

Impact on the System

- After the service attributes are configured, the service needs to be restarted. The service cannot be accessed during restart.
- You need to download and update the client configuration files after configuring HBase, HDFS, Hive, Spark, Yarn, and MapReduce service properties.

Prerequisites

- You have understood the meanings of parameters to be added, configuration files that have taken effect, and the impact on components.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Select the target service from the service list.

Step 3 Click **Service Configuration**.

Step 4 In the configuration type drop-down box on the right side, switch **Basic** to **All**.

Step 5 In the navigation tree, select **Customization**. The customized parameters of the current component are displayed on MRS.

The configuration files that save the newly added customized parameters are displayed in the **Parameter File** column. Different configuration files may have same open-source parameters. After the parameters in different files are set to different values, whether the configuration takes effect depends on the loading sequence of the configuration files by components. You can customize parameters for services and roles as required. Adding customized parameters for a single role instance is not supported.

Step 6 Based on the configuration files and parameter functions, locate the row where a specified parameter resides, enter the parameter name supported by the component in the **Parameter** column and enter the parameter value in the **Value** column.

- You can click ⊕ or ⊕ to add or delete a customized parameter. You can delete a customized parameter only after you click ⊕ for the first time.
- If you want to cancel the modification of a parameter value, click ↻ to restore it.

Step 7 Click **Save Configuration**, select **Restart the affected services or instances**, and click **OK**.

----End

Task Example

Configuring Customized Hive Parameters

Hive depends on HDFS. By default, Hive accesses the HDFS client. The configuration parameters to take effect are controlled by HDFS in a unified manner. For example, the HDFS parameter **ipc.client.rpc.timeout** affects the RPC timeout period for all clients to connect to the HDFS server. If you need to modify the timeout period for Hive to connect to HDFS, you can use the configuration customization function. After this parameter is added to the **core-site.xml** file of Hive, this parameter can be identified by the Hive service and its configuration overwrites the parameter configuration in HDFS.

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Choose **Hive** > **Service Configuration**.

Step 3 In the configuration type drop-down box on the right side, switch **Basic** to **All**.

Step 4 In the navigation tree on the left, select **Customization** for the Hive service. The system displays the customized service parameters supported by Hive.

Step 5 In **core-site.xml**, locate the row that contains the **core.site.customized.configs** parameter, enter **ipc.client.rpc.timeout** in the **Parameter** column, and enter a new value in the **Value** column, for example, **150000**. The unit is millisecond.

Step 6 Click **Save Configuration**, select **Restart the affected services or instances**, and click **OK**.

Operation successful is displayed. Click **Finish**. The service is started successfully.

----End

5.6.6 Synchronizing Service Configuration

Scenario

If **Configuration Status** of some services is **Configuration expired** or **Configuration failed**, synchronize configuration for the cluster or service to restore its configuration status. If all services in the cluster are in the **Configuration failed** state, synchronize the cluster configuration with the background configuration.

Impact on the System

After synchronizing service configurations, you need to restart the services whose configurations have expired. These services are unavailable during restart.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Select the target service from the service list.

Step 3 On the Service Status tab page, choose **More** > **Synchronize Configuration**.

Step 4 In the dialog box that is displayed, select **Restart the service or instances whose configurations have expired** and click **Yes** to restart the service.

----End

5.6.7 Managing Role Instances

Scenario

You can start a role instance that is in the **Stopped**, **Failed to stop** or **Failed to start** status, stop an unused or abnormal role instance or restart an abnormal role instance to recover its functions.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Select the target service from the service list.

Step 3 Click the **Instances** tab.

Step 4 Select the check box on the left of the target role instance.

Step 5 Click **More**, select operations such as **Start Instance**, **Stop Instance**, **Restart Instance**, **Rolling-restart Instance**, or **Delete Instance** based on site requirements.

----End

5.6.8 Configuring Role Instance Parameters

Scenario

You can view and modify default role instance configuration on MRS based on site requirements. The configurations can be imported and exported.

Impact on the System

You need to download and update the client configuration files after configuring HBase, HDFS, Hive, Spark, Yarn, and MapReduce service properties.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Modifying Role Instance Parameters

1. On the MRS cluster details page, click **Components**.
2. Select the target service from the service list.
3. Click the **Instances** tab.
4. Click the target role instance from the role instance list.
5. Click the **Instance Configuration** tab.
6. Switch **Basic** to **All** from the drop-down list on the right of the page. All configuration parameters of the role instance are displayed in the navigation tree.
7. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

   If you want to cancel the modification of a parameter value, click  to restore it.
8. Click **Save Configuration**, select **Restart the affected services or instances**, and click **OK**.

5.6.9 Synchronizing Role Instance Configuration

Scenario

When **Configuration Status** of a role instance is **Configuration expired** or **Configuration failed**, you can synchronize the configuration data of the role instance with the background configuration.

Impact on the System

After synchronizing a role instance configuration, you need to restart the role instance whose configuration has expired. The role instance is unavailable during restart.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Select a service name.

Step 3 Click the **Instances** tab.

Step 4 Click the target role instance from the role instance list.

Step 5 Choose **More** > **Synchronize Configuration** above the role instance status and indicator information.

Step 6 In the dialog box that is displayed, select **Restart the service or instances whose configurations have expired** and click **Yes** to restart the role instance.

----End

5.6.10 Decommissioning and Recommissioning a Role Instance

Scenario

If a Core or Task node is faulty, the cluster status may be displayed as **Abnormal**. In an MRS cluster, data can be stored on different Core nodes. You can decommission the specified role instance on MRS to stop the role instance from providing services. After fault rectification, you can recommission the role instance.

The following role instances can be decommissioned or recommissioned:

- DataNode role instance on HDFS
- NodeManager role instance on Yarn
- RegionServer role instance on HBase
- ClickHouseServer role instance on ClickHouse
- Broker role instance on Kafka

Restrictions:

- If the number of the DataNodes is less than or equal to that of HDFS copies, decommissioning cannot be performed. If the number of HDFS copies is three and the number of DataNodes is less than four in the system, decommissioning cannot be performed. In this case, an error will be reported and force MRS to exit the decommissioning 30 minutes after MRS attempts to perform the decommissioning.
- If the number of Kafka Broker instances is less than or equal to that of Kafka copies, decommissioning cannot be performed. For example, if the number of Kafka copies is two and the number of nodes is less than three in the system, decommissioning cannot be performed. Instance decommissioning will fail and exit.
- If a role instance is out of service, you must recommission the instance to start it before using it again.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS cluster details page, click **Components**.

Step 2 Click a service in the service list.

Step 3 Click the **Instances** tab.

Step 4 Select an instance.

Step 5 Choose **More** > **Decommission** or **Recommission** to perform the corresponding operation.

NOTE

During the instance decommissioning, if the service corresponding to the instance is restarted in the cluster using another browser, MRS displays a message indicating that the instance decommissioning is stopped, but the **Operating Status** of the instance is displayed as **Started**. In this case, the instance has been decommissioned on the background. You need to decommission the instance again to synchronize the operating status.

----End

5.6.11 Starting and Stopping a Cluster

A cluster is a collection of service components. You can start or stop all services in a cluster.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

On the cluster details page, choose **Management Operations** > **Start All Components** or **Stop All Components** in the upper right corner to perform the required operation.

5.6.12 Synchronizing Cluster Configuration

Scenario

If **Configuration Status** of all services or some services is **Configuration expired** or **Configuration failed**, synchronize configuration for the cluster or service to restore its configuration status.

- If all services in the cluster are in the **Configuration failed** status, synchronize the cluster configuration with the background configuration.
- If all services in the cluster are in the **Configuration failed** status, synchronize the service configuration with the background configuration.

NOTE

In **MRS 3.x**, you cannot perform operations in this section on the management console.

Impact on the System

After synchronizing cluster configurations, you need to restart the services whose configurations have expired. These services are unavailable during restart.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the cluster details page, choose **Configuration** > **Synchronize Configuration** in the upper right corner.

Step 2 In the displayed dialog box, select **Restart services and instances whose configuration have expired**, and click **OK** to restart the service whose configuration has expired.

When **Operation successful** is displayed, click **Finish**. The cluster is started successfully.

----End

5.6.13 Exporting Cluster Configuration

Scenario

You can export all configuration data of a cluster using MRS to meet site requirements. The exported configuration data is used to rapidly update service configuration.

NOTE

In **MRS 3.x**, you cannot perform operations in this section on the management console.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

On the cluster details page, choose **Configuration** > **Export Cluster Configuration** in the upper right corner.

The exported file is used to update service configurations. For details, see **Importing Service Configuration Parameters** in **Configuring Service Parameters**.

5.6.14 Performing Rolling Restart

After modifying the configuration items of a big data component, you need to restart the corresponding service to make new configurations take effect. If you use a normal restart mode, all services or instances are restarted concurrently, which may cause service interruption. To ensure that services are not affected during service restart, you can restart services or instances in batches by rolling restart. For instances in active/standby mode, a standby instance is restarted first and then an active instance is restarted. Rolling restart takes longer than normal restart.

Table 5-49 provides services and instances that support or do not support rolling restart in the MRS cluster.

Table 5-49 Services and instances that support or do not support rolling restart

Service	Instance	Whether to Support Rolling Restart
HDFS	NameNode	Yes
	Zkfc	
	JournalNode	
	HttpFS	
	DataNode	
Yarn	ResourceManager	Yes
	NodeManager	
Hive	MetaStore	Yes
	WebHCat	
	HiveServer	
Mapreduce	JobHistoryServer	Yes
HBase	HMaster	Yes
	RegionServer	
	ThriftServer	
	RESTServer	
Spark	JobHistory	Yes
	JDBCServer	
	SparkResource	No
Hue	Hue	No
Tez	TezUI	No
Loader	Sqoop	No
Zookeeper	Quorumpeer	Yes
Kafka	Broker	Yes
	MirrorMaker	No
Flume	Flume	Yes
	MonitorServer	
Storm	Nimbus	Yes

Service	Instance	Whether to Support Rolling Restart
	UI	
	Supervisor	
	Logviewer	

Restrictions

- Perform a rolling restart during off-peak hours.
  - Otherwise, a rolling restart failure may occur. For example, if the throughput of Kafka is high (over 100 MB/s) during the Kafka rolling restart, the Kafka rolling restart may fail.
  - For example, if the requests per second of each RegionServer on the native interface exceed 10,000 during the HBase rolling restart, you need to increase the number of handles to prevent a RegionServer restart failure caused by heavy loads during the restart.
- Before the restart, check the number of current requests of HBase. If the number of requests of each RegionServer on the native interface exceeds 10,000, increase the number of handles to prevent a failure.
- If the number of Core nodes in a cluster is less than six, services may be affected for a short period of time.
- Preferentially perform a rolling instance or service restart and select **Only restart instances whose configurations have expired**.

Performing a Rolling Service Restart

Step 1 Choose **Clusters** > **Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click **Components** and select a service for which you want to perform a rolling restart.

Step 3 On the **Service Status** tab page, click **More** and select **Rolling-restart Service**.

Step 4 The **Rolling-restart Service** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the service.

Step 5 After the rolling restart task is complete, click **Finish**.

----End

Performing a Rolling Instance Restart

Step 1 Choose **Clusters** > **Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click **Components** and select a service for which you want to perform a rolling restart.

Step 3 On the **Instance** tab page, select the instance to be restarted. Click **More** and select **Rolling-restart Instance**.

Step 4 After you enter the administrator password, the **Rolling-restart Instance** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the instance.

Step 5 After the rolling restart task is complete, click **Finish**.

----End

Perform a Rolling Cluster Restart

Step 1 Choose **Clusters** > **Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 In the upper right corner of the page, choose **Management Operations** > **Perform Rolling Cluster Restart**.

Step 3 The **Rolling-restart Cluster** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the cluster.

Step 4 After the rolling restart task is complete, click **Finish**.

----End

Rolling Restart Parameter Description

Table 5-50 describes rolling restart parameters.

Table 5-50 Rolling restart parameter description

Parameter	Description
Only restart instances whose configurations have expired	Specifies whether to restart only the modified instances in a cluster.
Data Node Instances to Be Batch Restarted	Specifies the number of instances that are restarted in each batch when the batch rolling restart strategy is used. The default value is **1**. The value ranges from 1 to 20. This parameter is valid only for data nodes.
Batch Interval	Specifies the interval between two batches of instances for rolling restart. The default value is **0**. The value ranges from 0 to 2147483647. The unit is second. Note: Setting the batch interval parameter can increase the stability of the big data component process during the rolling restart. You are advised to set this parameter to a non-default value, for example, 10.

Parameter	Description
Batch Fault Tolerance Threshold	Specifies the tolerance times when the rolling restart of instances fails to be executed in batches. The default value is **0**, which indicates that the rolling restart task ends after any batch of instances fails to be restarted. The value ranges from 0 to 2147483647.

Procedure in a Typical Scenario

Step 1 Choose **Clusters** > **Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click **Components** and select **HBase**. The **HBase** service page is displayed.

Step 3 Click the **Service Configuration** tab, and modify an HBase parameter. After the following dialog box is displayed, click **OK** to save the configurations.

NOTE

Do not select **Restart the affected services or instances**. This option indicates a normal restart. If you select this option, all services or instances will be restarted, which may cause service interruption.

Step 4 After saving the configurations, click **Finish**.

Step 5 Click the **Service Status** tab.

Step 6 On the **Service Status** tab page, click **More** and select **Rolling-restart Service**.

Step 7 After you enter the administrator password, the **Rolling-restart Service** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart.

Step 8 After the rolling restart task is complete, click **Finish**.

----End

5.7 Alarm Management

5.7.1 Viewing the Alarm List

The alarm list displays all alarms in the MRS cluster. The MRS page displays the alarms that need to be handled in a timely manner and the events.

On the MRS management console, you can only query basic information about uncleared MRS alarms on the **Alarms** tab page. For details about how to view alarm details or manage alarms, see **Viewing and Manually Clearing an Alarm**.

Alarms are listed in chronological order by default in the alarm list, with the most recent alarms displayed at the top.

Table 5-51 describes various fields in an alarm.

Table 5-51 Alarm description

Parameter	Description
Alarm ID	ID of an alarm.
Alarm Name	Name of an alarm.

Parameter	Description
Severity	Alarm severity. In versions earlier than MRS 3.x, the cluster alarm severity is as follows: • Critical Indicates alarms reporting errors that affect cluster running, such as unavailable cluster services, node faults, data inconsistency between the active and standby GaussDB databases, and abnormal LdapServer data synchronization. You need to check the cluster status based on the alarms and rectify the faults in a timely manner. • Major Indicates alarms reporting errors that affect some cluster functions, including process faults, periodic backup task failures, and abnormal key file permissions. Check the objects for which the alarms are generated based on the alarms and clear the alarms in a timely manner. • Minor Indicates alarms reporting errors that do not affect major functions of the current cluster, including alarms indicating that the certificate file is about to expire, audit logs fail to be dumped, and the license file is about to expire. • Warning Indicates an alarm of the lowest severity. It is used for information display or prompt and indicates that an event occurs in the scenarios when you stop a service, delete a service, stop an instance, delete an instance, delete a node, restart a service, restart an instance, perform an active/standby switchover for MRS Manager, scale in a host, or restore an instance. Additionally, this type of alarms also occurs when an instance is faulty, a job executed successfully, or a job failed to be executed. In MRS 3.x or later, the alarm severity of a cluster is as follows: • Critical Indicates alarms reporting errors that affect cluster running, such as unavailable cluster services, node faults, data inconsistency between the active and standby GaussDB databases, and abnormal LdapServer data synchronization. You need to check the cluster status based on the alarms and rectify the faults in a timely manner. • Major Indicates alarms reporting errors that affect some cluster functions, including process faults, periodic backup task failures, and abnormal key file permissions. Check the objects for which the alarms are generated based on the alarms and clear the alarms in a timely manner. • Minor Indicates alarms reporting errors that do not affect major functions of the current cluster, including alarms indicating

Parameter	Description
	that the certificate file is about to expire, audit logs fail to be dumped, and the license file is about to expire. • Suggestion Indicates an alarm of the lowest severity. It is used for information display or prompt and indicates that an event occurs in the scenarios when you stop a service, delete a service, stop an instance, delete an instance, delete a node, restart a service, restart an instance, perform an active/standby switchover for MRS Manager, scale in a host, or restore an instance. Additionally, this type of alarms also occurs when an instance is faulty, a job executed successfully, or a job failed to be executed.
Generated	Time when the alarm is generated.
Location	Details about the alarm.
Operation	If the alarm can be manually cleared, click **Clear Alarm**.

Table 5-52 Button description

Button	Description
	Select an interval for refreshing the alarm list from the drop-down list. • Refresh every 30s • Refresh every 60s • Stop refreshing
	Select an alarm severity from the drop-down list box to filter alarms. For versions earlier than MRS 3.x, the following alarms can be filtered: All, Critical, Major, Minor, and Warning. (For MRS 3.x or later) You can filter the following alarms: All, Critical, Major, Minor, and Warning.
	Click  and manually refresh the alarm list.
Advanced Search	Click **Advanced Search**. In the displayed alarm search area, set search criteria and click **Search** to view the information about specified alarms. You can click **Reset** to clear the search criteria.

5.7.2 Viewing the Event List

The event list displays information about all events in a cluster, such as service restart and service termination.

Events are listed in chronological order by default in the event list, with the most recent events displayed at the top.

Table 5-53 describes various fields for an event.

Table 5-53 Event description

Parameter	Description
Event ID	Specifies the ID of an event.
Event Severity	Specifies the event severity. In versions earlier than MRS 3.x, the cluster event level is as follows: • Critical • Major • Minor • Suggestion In MRS 3.x or later, the event level of a cluster is as follows: • Critical • Major • Minor • Suggestion
Event Name	Name of the generated event.
Generated	Time when the event is generated.
Location	Specifies the detailed information for locating the event,

Table 5-54 Icon description

Icon	Description
	Select an interval for refreshing the event list from the drop-down list. • Refresh every 30s • Refresh every 60s • Stop refreshing
	Click  to manually refresh the event list.
Advanced Search	Click **Advanced Search**. In the displayed event search area, set search criteria and click **Search** to view the information about specified events. Click **Reset** to clear the search criteria.

Exporting events

Step 1 Choose **Clusters > Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click **Alarm Management** > **Events**.

Step 3 Click **Export All**.

Step 4 In the displayed dialog box, select the type and click **OK**.

----End

Common Events

Table 5-55 Common events

Event ID	Event Name
12019	Stop Service
12020	Delete Service
12021	Stop RoleInstance
12022	Delete RoleInstance
12023	Delete Node
12024	Restart Service
12025	Restart RoleInstance
12026	Manager Switchover
12065	Process Restart
12070	Job Running Succeeded
12071	Job Running Failed
12072	Job killed
12086	Agent Restart
14005	NameNode Switchover
14028	HDFS DiskBalancer Task
14029	Active NameNode enters safe mode and generates new Fsimage
17001	Oozie Workflow Execution Failure
17002	Oozie Scheduled Job Execution Failure
18001	ResourceManager Switchover
18004	JobHistoryServer Switchover

Event ID	Event Name
19001	HMaster Failover
20003	Hue Failover
24002	Flume Channel Overflow
25001	LdapServer Failover
27000	DBServer Switchover
38003	Adjusts the topic data storage period
43014	Spark Data Skew
43015	Spark SQL Large Query Results
43016	Spark SQL Execution Timeout
43024	Start JDBCServer
43025	Stop JDBCServer
43026	ZooKeeper Connection Succeeded
43027	Zookeeper Connection Failed

5.7.3 Viewing and Manually Clearing an Alarm

Scenario

You can view and clear alarms on MRS.

Generally, the system automatically clears an alarm when the fault is rectified. If the fault has been rectified and the alarm cannot be automatically cleared, you can manually clear the alarm.

You can view the latest 100,000 alarms (including uncleared, manually cleared, and automatically cleared alarms) on MRS. If the number of cleared alarms exceeds 100,000 and is about to reach 110,000, the system automatically dumps the earliest 10,000 cleared alarms to the dump path.

3. In versions earlier than x, the value is the same as that of ${BIGDATA_HOME}/OMSV100R001C00x8664/workspace/data for the active management node.

(For 3.x and later versions) The path is **${BIGDATA_HOME}/om-server/OMS/workspace/data** of the active management node.

A directory is automatically generated when alarms are dumped for the first time.

NOTE

Set an automatic refresh interval or click  for an immediate refresh.

The following refresh interval options are supported:

- Refresh every 30 seconds
- Refresh every 60 seconds
- Stop refreshing

Procedure

Step 1 Choose **Clusters > Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click **Alarms** and view the alarm information in the alarm list.

- By default, the alarm list page displays the latest 10 alarms.
- By default, data is sorted in descending order based on the generation time. For MRS 3.x or earlier, you can click the alarm ID, severity, and generation time to modify the sorting mode. For clusters of MRS 3.x or later, you can click the severity and generation time to modify the sorting mode.
- You can filter all alarms of the same severity. The results include cleared and uncleared alarms.
- For clusters of MRS 3.x and earlier versions, you can click  ,  , or  in the upper right corner of the page to quickly filter **Critical**, **Major**, **Minor**, or **Suggestion** alarms that are uncleared.
- For clusters of MRS 3.x or later: You can click  ,  , or  in the upper right corner of the page to quickly filter uncleared **Critical**, **Major**, **Minor** or **Warning** alarms.

Step 3 Click **Advanced Search**. In the displayed alarm search area, set search criteria and click **Search** to view the information about specified alarms. You can click **Reset** to clear the search criteria.

NOTE

The start time and end time are specified in **Time Range**. You can search for alarms generated within the time range.

Handle the alarm by referring to **Alarm Reference**. If the alarms in some scenarios are generated due to other cloud services that MRS depends on, you need to contact maintenance personnel of the corresponding cloud services.

Step 4 If the alarm needs to be manually cleared after errors are rectified, click **Clear Alarm**.

NOTE

If multiple alarms have been handled, you can select one or more alarms to be cleared and click **Clear Alarm** to clear the alarms in batches. A maximum of 300 alarms can be cleared in each batch.

----End

Exporting Alarms

Step 1 Choose **Clusters > Active Clusters** and click a cluster name to go to the cluster details page.

Step 2 Click **Alarm Management** > **Alarms**.

Step 3 Click **Export All**.

Step 4 In the displayed dialog box, select the type and click **OK**.

----End

5.8 Patch Management

5.8.1 Patch Operation Guide for Versions Earlier Than MRS 3.x

If you obtain patch information from the following sources, upgrade the patch according to actual requirements.

- You obtain information about the patch released by MRS from a message pushed by the message center service.
- You obtain information about the patch by accessing the cluster and viewing patch information.

Preparing for Patch Installation

- Follow instructions in **Performing a Health Check** to check cluster status. If the cluster health status is normal, install a patch.
- You need to confirm the target patch to be installed according to the patch information in the patch content.

Installing a Patch

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patches** tab page, click **Install** in the **Operation** column to install the target patch.

NOTE

- For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

Uninstalling a Patch

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patches** page, click **Uninstall** in the **Operation** column to uninstall the target patch.

NOTE

- For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

5.8.2 Rolling Patches

The rolling patch function indicates that patches are installed or uninstalled for one or more services in a cluster by performing a rolling service restart (restarting services or instances in batches), without interrupting the services or within a minimized service interruption interval. Services in a cluster are divided into the following three types based on whether they support rolling patch:

- Services supporting rolling patch installation or uninstallation: All businesses or part of them (varying depending on different services) of the services are not interrupted during patch installation or uninstallation.
- Services not supporting rolling patch installation or uninstallation: Businesses of the services are interrupted during patch installation or uninstallation.
- Services with some roles supporting rolling patch installation or uninstallation: Some businesses of the services are not interrupted during patch installation or uninstallation.

NOTE

In **MRS 3.x**, you cannot perform operations in this section on the management console.

Table 5-56 provides services and instances that support or do not support rolling restart in the MRS cluster.

Table 5-56 Services and instances that support or do not support rolling restart

Service	Instance	Whether to Support Rolling Restart
HDFS	NameNode	Yes
	Zkfc	
	JournalNode	
	HttpFS	
	DataNode	
Yarn	ResourceManager	Yes
	NodeManager	
Hive	MetaStore	Yes
	WebHCat	
	HiveServer	

Service	Instance	Whether to Support Rolling Restart
MapReduce	JobHistoryServer	Yes
HBase	HMaster	Yes
	RegionServer	
	ThriftServer	
	RESTServer	
Spark	JobHistory	Yes
	JDBCServer	
	SparkResource	No
Hue	Hue	No
Tez	TezUI	No
Loader	Sqoop	No
Zookeeper	Quorumpeer	Yes
Kafka	Broker	Yes
	MirrorMaker	No
Flume	Flume	Yes
	MonitorServer	
Storm	Nimbus	Yes
	UI	
	Supervisor	
	LogViewer	

Installing a Patch

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patches** page, click **Install** in the **Operation** column.

Step 4 On the **Warning** page, enable or disable **Rolling Patch**.

NOTE

- Enabling the rolling patch installation function: Services are not stopped before patch installation, and rolling service restart is performed after the patch installation. This minimizes the impact on cluster services but takes more time than common patch installation.
- Disabling the rolling patch uninstallation function: All services are stopped before patch uninstallation, and all services are restarted after the patch uninstallation. This temporarily interrupts the cluster and the services but takes less time than rolling patch uninstallation.
- The rolling patch installation function is not available in clusters with less than two Master nodes and three Core nodes.

Step 5 Click **Yes** to install the target patch.

Step 6 View the patch installation progress.

1. Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.
2. Choose **System** > **Manage Patch**. On the **Manage Patch** page, you can view the patch installation progress.

NOTE

For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

Uninstalling a Patch

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patches** page, click **Uninstall** in the **Operation** column.

Step 4 On the **Warning** page, enable or disable **Rolling Patch**.

NOTE

- Enabling the rolling patch uninstallation function: Services are not stopped before patch uninstallation, and rolling service restart is performed after the patch uninstallation. This minimizes the impact on cluster services but takes more time than common patch uninstallation.
- Disabling the rolling patch uninstallation function: All services are stopped before patch uninstallation, and all services are restarted after the patch uninstallation. This temporarily interrupts the cluster and the services but takes less time than rolling patch uninstallation.
- Only patches that are installed in rolling mode can be uninstalled in the same mode.

Step 5 Click **Yes** to uninstall the target patch.

Step 6 View the patch uninstallation progress.

1. Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.
2. Choose **System** > **Manage Patch**. On the **Manage Patch** page, you can view the patch uninstallation progress.

NOTE

For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

5.8.3 Restoring Patches for the Isolated Hosts

If some hosts are isolated in a cluster, perform the following operations to restore patches for these isolated hosts after patch installation on other hosts in the cluster. After patch restoration, versions of the isolated host nodes are consistent with those are not isolated.

NOTE

In **MRS 3.x**, you cannot perform operations in this section on the management console.

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 Choose **System > Manage Patch**. The **Manage Patch** page is displayed.

Step 3 In the **Operation** column, click **View Details**.

Step 4 On the patch details page, select host nodes whose **Status** is **Isolated**.

Step 5 Click **Select and Restore** to restore the isolated host nodes.

----End

5.9 Tenant Management

5.9.1 Before You Start

This section describes how to manage tenants on the MRS console.

Tenant management operations on the console apply only to clusters of versions earlier than MRS 3.x.

Tenant management operations on FusionInsight Manager apply to all versions. For MRS 3.x and later versions, see **Overview**. For versions earlier than MRS 3.x, see **Overview**.

5.9.2 Overview

Definition

An MRS cluster provides various resources and services for multiple organizations, departments, or applications to share. The cluster provides tenants as a logical entity to use these resources and services. A mode involving different tenants is called multi-tenant mode. Currently, only the analysis cluster supports tenant management.

Principles

The MRS cluster provides the multi-tenant function. It supports a layered tenant model and allows dynamic adding or deleting of tenants to isolate resources. It dynamically manages and configures tenants' computing and storage resources.

The computing resources indicate tenants' Yarn task queue resources. The task queue quota can be modified, and the task queue usage status and statistics can be viewed.

The storage resources can be stored on HDFS. You can add and delete the HDFS storage directories of tenants, and set the quotas of file quantity and the storage space of the directories.

Tenants can create and manage tenants in a cluster based on service requirements.

- Roles, computing resources, and storage resources are automatically created when tenants are created. By default, all permissions of the new computing resources and storage resources are allocated to a tenant's roles.
- Permissions to view the current tenant's resources, add a subtenant, and manage the subtenant's resources are granted to the tenant's roles by default.
- After you have modified the tenant's computing or storage resources, permissions of the tenant's roles are automatically updated.

MRS supports a maximum of 512 tenants. The default tenants created by the system include **default**. Tenants that are in the topmost layer with the default tenant are called level-1 tenants.

Resource Pools

Yarn task queues support only the label-based scheduling policy. This policy enables Yarn task queues to associate NodeManagers that have specific node labels. In this way, Yarn tasks run on specified nodes so that tasks are scheduled and certain hardware resources are utilized. For example, Yarn tasks requiring a large memory capacity can run on nodes with a large memory capacity by means of label association, preventing poor service performance.

In an MRS cluster, the tenant logically divides Yarn cluster nodes to combine multiple NodeManagers into a resource pool. Yarn task queues can be associated with specified resource pools by configuring queue capacity policies, ensuring efficient and independent resource utilization in the resource pools.

MRS supports a maximum of 50 resource pools. By default, the system contains a **default** resource pool.

5.9.3 Creating a Tenant

Scenario

You can create a tenant on MRS Manager to specify the resource usage.

Prerequisites

- A tenant name has been planned. The name must not be the same as that of a role or Yarn queue that exists in the current cluster.

- If a tenant requires storage resources, a storage directory has been planned based on service requirements, and the planned directory does not exist under the HDFS directory.
- The resources that can be allocated to the current tenant have been planned and the sum of the resource percentages of direct sub-tenants under the parent tenant at every level does not exceed 100%.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click **Create Tenant**. On the page that is displayed, configure tenant properties.

Table 5-57 Tenant parameters

Parameter	Description
Name	Specifies the name of the current tenant. The value consists of 3 to 50 characters, and can contain letters, digits, and underscores (_).
Tenant Type	The options include **Leaf** and **Non-leaf**. If **Leaf** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. If **Non-leaf** is selected, sub-tenants can be added to the current tenant.
Dynamic Resource	Specifies the dynamic computing resources for the current tenant. The system automatically creates a task queue named after the tenant name in Yarn. When dynamic resources are not **Yarn**, the system does not automatically create a task queue.
Default Resource Pool Capacity (%)	Specifies the percentage of the computing resources used by the current tenant in the **default** resource pool.
Default Resource Pool Max. Capacity (%)	Specifies the maximum percentage of the computing resources used by the current tenant in the **default** resource pool.
Storage Resource	Specifies storage resources for the current tenant. The system automatically creates a file folder named after the tenant name in the **/tenant** directory. When a tenant is created for the first time, the system automatically creates the **/tenant** directory in the HDFS root directory. If storage resources are not **HDFS**, the system does not create a storage directory under the root directory of HDFS.

Parameter	Description
Space Quota (MB)	Specifies the quota for HDFS storage space used by the current tenant. The value ranges from **1** to **8796093022208**. The unit is MB. This parameter indicates the maximum HDFS storage space that can be used by a tenant, but does not indicate the actual space used. If the value is greater than the size of the HDFS physical disk, the maximum space available is the full space of the HDFS physical disk. **NOTE** To ensure data reliability, one backup is automatically generated for each file saved in HDFS, that is, two copies are generated in total. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).
Storage Path	Specifies the tenant's HDFS storage directory. The system automatically creates a file folder named after the tenant name in the **/tenant** directory by default. For example, the default HDFS storage directory for **ta1** is **tenant/ta1**. When a tenant is created for the first time, the system automatically creates the **/tenant** directory in the HDFS root directory. The storage path is customizable.
Service	Specifies other service resources associated with the current tenant. HBase is supported. To configure this parameter, click **Associate Services**. In the dialog box that is displayed, set **Service** to **HBase**. If **Association Mode** is set to **Exclusive**, service resources are occupied exclusively. If **share** is selected, service resources are shared.
Description	Specifies the description of the current tenant.

Step 3 Click **OK** to save the settings.

It takes a few minutes to save the settings. If the **Tenant created successfully** is displayed in the upper-right corner, the tenant is added successfully. The tenant is created successfully.

NOTE

- Roles, computing resources, and storage resources are automatically created when tenants are created.
- The new role has permissions on the computing and storage resources. The role and its permissions are controlled by the system automatically and cannot be controlled manually under **Manage Role**.
- If you want to use the tenant, create a system user and assign the Manager_tenant role and the role corresponding to the tenant to the user. For details, see **Creating a User**.

----End

Related Tasks

Viewing an added tenant

Step 1 On the MRS details page, click **Tenants**.

Step 2 In the tenant list on the left, click the name of the added tenant.

The **Summary** tab is displayed on the right by default.

Step 3 View **Basic Information**, **Resource Quota**, and **Statistics** of the tenant.

If HDFS is in the **Stopped** state, **Available** and **Used** of **Space** in **Resource Quota** are **unknown.**

----End

5.9.4 Creating a Sub-tenant

Scenario

You can create a sub-tenant on MRS if the resources of the current tenant need to be further allocated.

Prerequisites

- A parent tenant has been added.
- A tenant name has been planned. The name must not be the same as that of a role or Yarn queue that exists in the current cluster.
- If a sub-tenant requires storage resources, a storage directory has been planned based on service requirements, and the planned directory does not exist under the storage directory of the parent tenant.
- The resources that can be allocated to the current tenant have been planned and the sum of the resource percentages of direct sub-tenants under the parent tenant at every level does not exceed 100%.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 In the tenant list on the left, move the cursor to the tenant node to which a sub-tenant is to be added. Click **Create sub-tenant**. On the displayed page, configure the sub-tenant attributes according to the following table:

Table 5-58 Sub-tenant parameters

Parameter	Description
Parent tenant	Specifies the name of the parent tenant.

Parameter	Description
Name	Specifies the name of the current tenant. The value consists of 3 to 20 characters, and can contain letters, digits, and underscores (_).
Tenant Type	The options include **Leaf** and **Non-leaf**. If **Leaf** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. If **Non-leaf** is selected, sub-tenants can be added to the current tenant.
Dynamic Resource	Specifies the dynamic computing resources for the current tenant. The system automatically creates a task queue named after the sub-tenant name in the Yarn parent queue. When dynamic resources are not **Yarn**, the system does not automatically create a task queue. If the parent tenant does not have dynamic resources, the sub-tenant cannot use dynamic resources.
Default Resource Pool Capacity (%)	Specifies the percentage of the resources used by the current tenant. The base value is the total resources of the parent tenant.
Default Resource Pool Max. Capacity (%)	Specifies the maximum percentage of the computing resources used by the current tenant. The base value is the total resources of the parent tenant.
Storage Resource	Specifies storage resources for the current tenant. The system automatically creates a file in the HDFS parent tenant directory. The file is named the same as the name of the sub-tenant. If storage resources are not **HDFS**, the system does not create a storage directory under the root directory of HDFS. If the parent tenant does not have storage resources, the sub-tenant cannot use storage resources.

Parameter	Description
Space Quota (MB)	Specifies the quota for HDFS storage space used by the current tenant. The minimum value is 1, and the maximum value is the total storage quota of the parent tenant. The unit is MB. This parameter indicates the maximum HDFS storage space that can be used by a tenant, but does not indicate the actual space used. If the value is greater than the size of the HDFS physical disk, the maximum space available is the full space of the HDFS physical disk. If the quota is greater than the quota of the parent tenant, the actual storage capacity is subject to the quota of the parent tenant. **NOTE** To ensure data reliability, one backup is automatically generated for each file saved in HDFS, that is, two copies are generated in total. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).
Storage Path	Specifies the tenant's HDFS storage directory. The system automatically creates a file folder named after the sub-tenant name in the directory of the parent tenant by default. For example, if the sub-tenant is **ta1s** and the parent directory is **tenant/ta1**, the system sets this parameter for the sub-tenant to **tenant/ta1/ta1s**. The storage path is customizable in the parent directory. The parent directory for the storage path must be the storage directory of the parent tenant.
Service	Specifies other service resources associated with the current tenant. HBase is supported. To configure this parameter, click **Associate Services**. In the dialog box that is displayed, set **Service** to **HBase**. If **Association Mode** is set to **Exclusive**, service resources are occupied exclusively. If **share** is selected, service resources are shared.
Description	Specifies the description of the current tenant.

Step 3 Click **OK** to save the settings.

It takes a few minutes to save the settings. If the **Tenant created successfully** is displayed in the upper-right corner, the tenant is added successfully. The tenant is created successfully.

NOTE

- Roles, computing resources, and storage resources are automatically created when tenants are created.
- The new role has permissions on the computing and storage resources. The role and its permissions are controlled by the system automatically and cannot be controlled manually under **Manage Role**.
- When using this tenant, create a system user and assign the user a related tenant role. For details, see **Creating a User**.

----End

5.9.5 Deleting a Tenant

Scenario

You can delete a tenant that is not required on MRS.

Prerequisites

- A tenant has been added.
- You have checked whether the tenant to be deleted has sub-tenants. If the tenant has sub-tenants, delete them; otherwise, you cannot delete the tenant.
- The role of the tenant to be deleted cannot be associated with any user or user group. For details about how to cancel the binding between a role and a user, see **Modifying User Information**.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 In the tenant list on the left, move the cursor to the tenant node to be deleted and click **Delete**.

The **Delete Tenant** dialog box is displayed. If you want to save the tenant data, select **Reserve the data of this tenant**. Otherwise, the tenant's storage space will be deleted.

Step 3 Click **OK**.

It takes a few minutes to save the configuration. After the tenant is deleted successfully, the role and storage space of the tenant are also deleted.

NOTE

- After the tenant is deleted, the task queue of the tenant still exists in Yarn.
- If you choose not to reserve data when deleting the parent tenant, data of sub-tenants is also deleted if the sub-tenants use storage resources.

----End

5.9.6 Managing a Tenant Directory

Scenario

You can manage the HDFS storage directory used by a specific tenant on MRS. The management operations include adding a tenant directory, modifying the directory file quota, modifying the storage space, and deleting a directory.

Prerequisites

- A tenant associated with HDFS storage resources has been added.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

- View a tenant directory.
  a. On the MRS details page, click **Tenants**.

     NOTE

     For MRS 3.x or later, see **Overview**.
  b. In the tenant list on the left, click the target tenant.
  c. Click the **Resources** tab.
  d. View the **HDFS Storage** table.
     - The **Maximum Number of Files/Directories** column indicates the quotas for the file and directory quantity of the tenant directory.
     - The **Space Quota** column indicates storage space size of tenant directories.
- Add a tenant directory.
  a. On the MRS details page, click **Tenants**.

     NOTE

     For MRS 3.x or later, see **Overview**.
  b. In the tenant list on the left, click the tenant whose HDFS storage directory needs to be added.
  c. Click the **Resources** tab.
  d. In the **HDFS Storage** table, click **Create Directory**.
     - Set **Path** to a tenant directory path.

       NOTE

       - If the current tenant is not a sub-tenant, the new path is created in the HDFS root directory.
       - If the current tenant is a sub-tenant, the new path is created in the specified directory.

       A complete HDFS storage directory can contain a maximum of 1,023 characters. An HDFS directory name contains digits, letters, spaces, and underscores (_). The name cannot start or end with a space.

- Set **Maximum Number of Files/Directories** to the quotas of file and directory quantity.

  Maximum Number of Files/Directories is optional. Its value ranges from **1** to **9223372036854775806**.

- Set **Storage Space Quota** to the storage space size of the tenant directory.

  The value of **Storage Space Quota** ranges from **1** to **8796093022208**.

  NOTE

  To ensure data reliability, one backup is automatically generated for each file saved in HDFS, that is, two copies are generated in total. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value of **Storage Space Quota** is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).

e. Click **OK**. The system creates tenant directories in the HDFS root directory.

- Modify a tenant directory.

  a. On the MRS details page, click **Tenants**.

  NOTE

  For MRS 3.x or later, see **Overview**.

  b. In the tenant list on the left, click the tenant whose HDFS storage directory needs to be modified.

  c. Click the **Resources** tab.

  d. In the **HDFS Storage** table, click **Modify** in the **Operation** column of the specified tenant directory.

  - Set **Maximum Number of Files/Directories** to the quotas of file and directory quantity.

    Maximum Number of Files/Directories is optional. Its value ranges from **1** to **9223372036854775806**.

  - Set **Storage Space Quota** to the storage space size of the tenant directory.

    The value of **Storage Space Quota** ranges from **1** to **8796093022208**.

    NOTE

    To ensure data reliability, one backup is automatically generated for each file saved in HDFS, that is, two copies are generated in total. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value of **Storage Space Quota** is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).

  e. Click **OK**.

- Delete a tenant directory.

  a. On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

b. In the tenant list on the left, click the tenant whose HDFS storage directory needs to be deleted.

c. Click the **Resources** tab.

d. In the **HDFS Storage** table, click **Delete** in the **Operation** column of the specified tenant directory.

The default HDFS storage directory set during tenant creation cannot be deleted. Only the newly added HDFS storage directory can be deleted.

e. Click **OK**. The tenant directory is deleted.

5.9.7 Restoring Tenant Data

Scenario

Tenant data is stored on Manager and in cluster components by default. When components are restored from faults or reinstalled, some tenant configuration data may be abnormal. In this case, you can manually restore the tenant data.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 In the tenant list on the left, click a tenant node.

Step 3 Check the status of the tenant data.

1. In **Summary**, check the color of the circle on the left of **Basic Information**. Green indicates that the tenant is available and gray indicates that the tenant is unavailable.
2. Click **Resources** and check the circle on the left of **Yarn** or **HDFS Storage**. Green indicates that the resource is available, and gray indicates that the resource is unavailable.
3. Click **Service Association** and check the **Status** column of the associated service table. **Good** indicates that the component can provide services for the associated tenant. **Bad** indicates that the component cannot provide services for the tenant.
4. If any check result is abnormal, go to **Step 4** to restore tenant data.

Step 4 Click **Restore Tenant Data**.

Step 5 In the **Restore Tenant Data** window, select one or more components whose data needs to be restored. Click **OK**. The system automatically restores the tenant data.

----End

5.9.8 Creating a Resource Pool

Scenario

In an MRS cluster, users can logically divide Yarn cluster nodes to combine multiple NodeManagers into a Yarn resource pool. Each NodeManager belongs to one resource pool only. The system contains a **default** resource pool by default. All NodeManagers that are not added to customized resource pools belong to this resource pool.

You can create a customized resource pool on MRS and add hosts that have not been added to other customized resource pools to it.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click the **Resource Pools** tab.

Step 3 Click **Create Resource Pool**.

Step 4 In **Create Resource Pool**, set the properties of the resource pool.

- **Name**: Enter a name for the resource pool. The name of the newly created resource pool cannot be **default**.

  The name consists of 1 to 20 characters and can contain digits, letters, and underscores (_) but cannot start with an underscore (_).

- **Available Hosts**: In the host list on the left, select a specified host name and add it to the resource pool. Only hosts in the cluster can be selected. The host list of a resource pool can be left blank.

Step 5 Click **OK**.

Step 6 After a resource pool is created, users can view the **Name**, **Members**, **Type**, **vCore** and **Memory** in the resource pool list. Hosts that are added to the customized resource pool are no longer members of the **default** resource pool.

----End

5.9.9 Modifying a Resource Pool

Scenario

You can modify members of an existing resource pool on MRS.

Prerequisites

You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click the **Resource Pools** tab.

Step 3 Locate the row that contains the specified resource pool, and click **Modify** in the **Operation** column.

Step 4 In **Modify Resource Pool**, modify **Added Hosts**.

- Adding a host: In the host list on the left, select the specified host name and add it to the resource pool.
- Deleting a host: In the host list on the right, click  next to a host to remove the host from the resource pool. The host list of a resource pool can be left blank.

Step 5 Click **OK**.

----End

5.9.10 Deleting a Resource Pool

Scenario

You can delete an existing resource pool on MRS.

Prerequisites

- Any queue in a cluster cannot use the resource pool to be deleted as the default resource pool. Before deleting the resource pool, cancel the default resource pool. For details, see **Configuring a Queue**.
- Resource distribution policies of all queues have been cleared from the resource pool being deleted. For details, see **Clearing Configuration of a Queue**.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenant**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click the **Resource Pools** tab.

Step 3 Locate the row that contains the specified resource pool, and click **Delete** in the **Operation** column.

In the displayed dialog box, click **OK**.

----End

5.9.11 Configuring a Queue

Scenario

You can modify the queue configuration of a specified tenant on MRS based on service requirements.

Prerequisites

- A tenant associated with Yarn and allocated dynamic resources has been added.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click the **Queue Configuration** tab.

Step 3 In the tenant queue table, click **Modify** in the **Operation** column of the specified tenant queue.

NOTE

- In the tenant list on the left of the **Tenant Management** tab, click the target tenant. In the window that is displayed, choose **Resource**. On the page that is displayed, click  to open the queue modification page.
- A queue can be bound to only one non-default resource pool.

Versions earlier than MRS 3.x:

Table 5-59 Queue configuration parameters

Parameter	Description
Maximum Applications	Specifies the maximum number of applications. The value ranges from 1 to 2147483647.
Maximum AM Resource Percent	Specifies the maximum percentage of resources that can be used to run the ApplicationMaster in a cluster. The value ranges from 0 to 1.
Minimum User Limit Percent (%)	Specifies the minimum percentage of resources consumed by a user. The value ranges from 0 to 100.
User Limit Factor	Specifies the limit factor of the maximum user resource usage. The maximum user resource usage percentage can be obtained by multiplying the limit factor with the percentage of the tenant's actual resource usage in the cluster. The minimum value is **0**.
Status	Specifies the current status of a resource plan. The values are **Running** and **Stopped**.
Default Resource Pool (Default Node Label Expression)	Specifies the resource pool used by a queue. The default value is **default**. If you want to change the resource pool, configure the queue capacity first. For details, see **Configuring the Queue Capacity Policy of a Resource Pool**.

MRS 3.x or later:

Table 5-60 Queue configuration parameters

Parameter	Description
Max Master Shares (%)	Indicates the maximum percentage of resources occupied by all ApplicationMasters in the current queue.
Max Allocated vCores	Indicates the maximum number of cores that can be allocated to a single YARN container in the current queue. The default value is **-1**, indicating that the number of cores is not limited within the value range.
Max Allocated Memory (MB)	Indicates the maximum memory that can be allocated to a single Yarn container in the current queue. The default value is **-1**, indicating that the memory is not limited within the value range.

Parameter	Description
Max Running Apps	Maximum number of tasks that can be executed at the same time in the current queue. The default value is **-1**, indicating that the number is not limited within the value range (the meaning is the same if the value is empty). The value 0 indicates that the task cannot be executed. The value ranges from -1 to 2147483647.
Max Running Apps per User	Maximum number of tasks that can be executed by each user in the current queue at the same time. The default value is **-1**, indicating that the number is not limited within the value range. If the value is **0**, the task cannot be executed. The value ranges from -1 to 2147483647.
Max Pending Apps	Maximum number of tasks that can be suspended at the same time in the current queue. The default value is **-1**, indicating that the number is not limited within the value range (the meaning is the same if the value is empty). The value **0** indicates that tasks cannot be suspended. The value ranges from -1 to 2147483647.
Resource Allocation Rule	Indicates the rule for allocating resources to different tasks of a user. The rule can be FIFO or FAIR. If a user submits multiple tasks in the current queue and the rule is FIFO, the tasks are executed one by one in sequential order. If the rule is FAIR, resources are evenly allocated to all tasks.
Default Resource Label	Indicates that tasks are executed on a node with a specified resource label. **NOTE** If you need to use a new resource pool, change the default label to the new resource pool label.
Active	• **ACTIVE**: indicates that the current queue can receive and execute tasks. • **INACTIVE**: indicates that the current queue can receive but cannot execute tasks. Tasks submitted to the queue are suspended.
Open	• **OPEN**: indicates that the current queue is opened. • **CLOSED**: indicates that the current queue is closed. Tasks submitted to the queue are rejected.

----End

5.9.12 Configuring the Queue Capacity Policy of a Resource Pool

Scenario

After a resource pool is added, the capacity policies of available resources need to be configured for Yarn task queues. This ensures that tasks in the resource pool are running properly. Each queue can be configured with the queue capacity policy of only one resource pool. Users can view the queues in any resource pool and configure queue capacity policies. After the queue policies are configured, Yarn task queues and resource pools are associated.

You can configure queue policies on MRS.

Prerequisites

- A resource pool has been added.
- The task queues are not associated with other resource pools. By default, all queues are associated with the **default** resource pool.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click the **Resource Distribution Policies** tab.

Step 3 In **Resource Pools**, select a specified resource pool.

Available Resource Quota: indicates that all resources in each resource pool are available for queues by default.

Step 4 Locate the specified queue in the **Resource Allocation** table, and click **Modify** in the **Operation** column.

Step 5 In **Modify Resource Allocation**, configure the resource capacity policy of the task queue in the resource pool.

- **Capacity (%)**: specifies the percentage of the current tenant's computing resource usage.
- **Maximum Capacity (%)**: specifies the percentage of the current tenant's maximum computing resource usage.

Step 6 Click **OK** to save the settings.

----End

5.9.13 Clearing Configuration of a Queue

Scenario

Users can clear the configuration of a queue on MRS Manager when the queue does not need resources from a resource pool or if a resource pool needs to be disassociated from the queue. Clearing queue configurations means that the resource capacity policy of the queue is canceled.

Prerequisites

- If a queue is to be unbound from a resource pool, this resource pool cannot serve as the default resource pool of the queue. Therefore, you must first change the default resource pool of the queue to another one. For details, see **Configuring a Queue**.
- You have synchronized IAM users. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Procedure

Step 1 On the MRS details page, click **Tenants**.

NOTE

For MRS 3.x or later, see **Overview**.

Step 2 Click the **Resource Distribution Policies** tab.

Step 3 In **Resource Pools**, select a specified resource pool.

Step 4 Locate the specified queue in the **Resource Allocation** table, and click **Clear** in the **Operation** column

In the **Clear Queue Configuration** dialog box, click **OK** to clear the queue configuration in the current resource pool.

NOTE

If no resource capacity policy is configured for a queue, the clearing function is unavailable for the queue by default.

----End

6 Using an MRS Client

6.1 Installing a Client

6.1.1 Installing a Client (Version 3.x or Later)

Scenario

This section describes how to install clients of all services (excluding Flume) in an MRS cluster. For details about how to install the Flume client, see "Using Flume" > "Installing the Flume Client" in *MapReduce Service Component Operation Guide*.

A client can be installed on a node inside or outside the cluster. This section uses the installation directory **//opt/client** as an example. Replace it with the actual one.

Prerequisites

- A Linux ECS has been prepared. For details about the supported OS of the ECS, see **Table 6-1**.

Table 6-1 Reference list

CPU Architecture	OS	Supported Version
x86 computing	Euler	Euler OS 2.5
	SuSE	SUSE Linux Enterprise Server 12 SP4 (SUSE 12.4)
	Red Hat	Red Hat-7.5-x86_64 (Red Hat 7.5)
	CentOS	CentOS 7.6

CPU Architecture	OS	Supported Version
Kunpeng computing (Arm)	Euler	Euler OS 2.8
	CentOS	CentOS 7.6

In addition, sufficient disk space is allocated for the ECS, for example, 40 GB.

- The ECS and the MRS cluster are in the same VPC.
- The security group of the ECS must be the same as that of the master node in the MRS cluster.
- The NTP service has been installed on the ECS OS and is running properly.

  If the NTP service is not installed, run the **yum install ntp -y** command to install it when the **yum** source is configured.
- A user can log in to the Linux ECS using the password (in SSH mode).

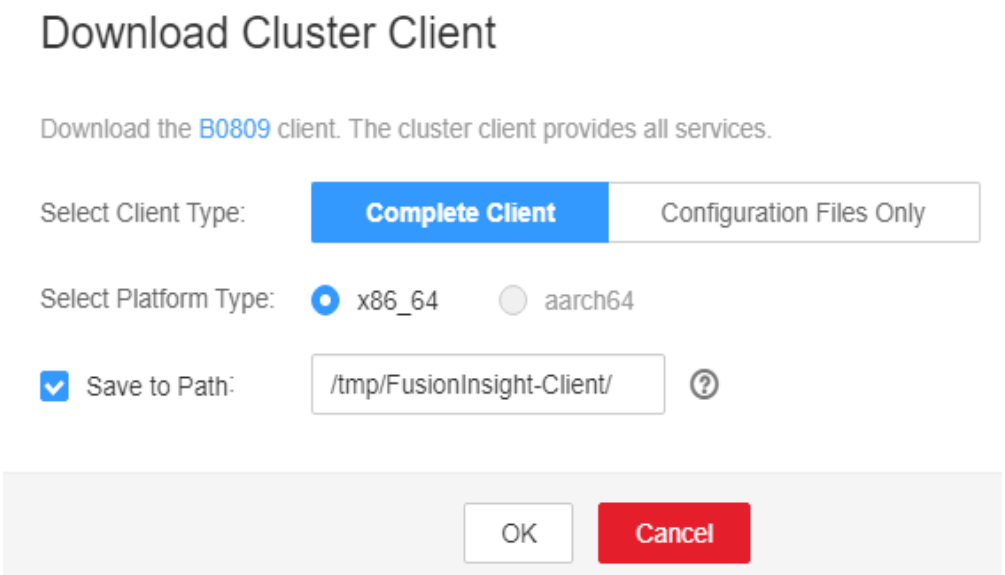
Installing a Client on a Node Inside a Cluster

1. Obtain the software package.

   Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Click the name of the cluster to be operated in the **Cluster** drop-down list.

   Choose **More > Download Client**. The **Download Cluster Client** dialog box is displayed.

   Figure 6-1 Downloading a client



NOTE

In the scenario where only one client is to be installed, choose **Cluster > Service >** *Service name* **> More > Download Client**. The **Download Client** dialog box is displayed.

2. Set the client type to **Complete Client**.

   Configuration Files Only is to download client configuration files in the following scenario: After a complete client is downloaded and installed and administrators modify server configurations on Manager, developers need to update the configuration files during application development.

   The platform type can be set to **x86_64** or **aarch64**.

   - **x86_64**: indicates the client software package that can be deployed on the x86 servers.
   - **aarch64**: indicates the client software package that can be deployed on the TaiShan servers.

   NOTE

   The cluster supports two types of clients: **x86_64** and **aarch64**. The client type must match the architecture of the node for installing the client. Otherwise, client installation will fail.

3. Select **Save to Path** and click **OK** to generate the client file.

   The generated file is stored in the **/tmp/FusionInsight-Client** directory on the active management node by default. You can also store the client file in a directory on which user **omm** has the read, write, and execute permissions. Copy the software package to the file directory on the server where the client is to be installed as user **omm** or **root**.

   The name of the client software package is in the follow format: **FusionInsight_Cluster_***<Cluster ID>***_Services_Client.tar**.

   The following steps and sections use **FusionInsight_Cluster_1_Services_Client.tar** as an example.

   NOTE

   If you cannot obtain the permissions of user **root**, use user **omm**.

   To install the client on another node in the cluster, run the following command to copy the client to the node where the client is to be installed:

   scp -p /*tmp/FusionInsight-Client***/FusionInsight_Cluster_1_Services_Client.tar** *IP address of the node where the client is to be installed:/opt/Bigdata/client*

4. Log in to the server where the client software package is located as user **user_client**.
5. Decompress the software package.

   Go to the directory where the installation package is stored, such as **/tmp/FusionInsight-Client**. Run the following command to decompress the installation package to a local directory:

   tar -xvf FusionInsight_Cluster_1_Services_Client.tar

6. Verify the software package.

   Run the following command to verify the decompressed file and check whether the command output is consistent with the information in the **sha256** file.

   sha256sum -c FusionInsight_Cluster_1_Services_ClientConfig.tar.sha256

```
FusionInsight_Cluster_1_Services_ClientConfig.tar: OK
```

7. Decompress the obtained installation file.

   tar -xvf FusionInsight_Cluster_1_Services_ClientConfig.tar

8. Go to the directory where the installation package is stored, and run the following command to install the client to a specified directory (an absolute path), for example, **/opt/client**:

   cd /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_ClientConfig

   Run the **./install.sh /opt/client** command to install the client. The client is successfully installed if information similar to the following is displayed:

   ```
   The component client is installed successfully
   ```

   NOTE

   - If the clients of all or some services use the **/opt/client** directory, other directories must be used when you install other service clients.
   - You must delete the client installation directory when uninstalling a client.
   - To ensure that an installed client can only be used by the installation user (for example, **user_client**), add parameter **-o** during the installation. That is, run the **./install.sh /opt/client -o** command to install the client.
   - If an HBase client is installed, it is recommended that the client installation directory contain only uppercase and lowercase letters, digits, and characters (_-?.@+=) due to the limitation of the Ruby syntax used by HBase.

Using a Client

1. On the node where the client is installed, run the **sudo su - omm** command to switch the user. Run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *MRS cluster user*

   Example: **kinit admin**

   NOTE

   User **admin** is created by default for MRS clusters with Kerberos authentication enabled and is used for administrators to maintain the clusters.

4. Run the client command of a component directly.

   For example, run the **hdfs dfs -ls /** command to view files in the HDFS root directory.

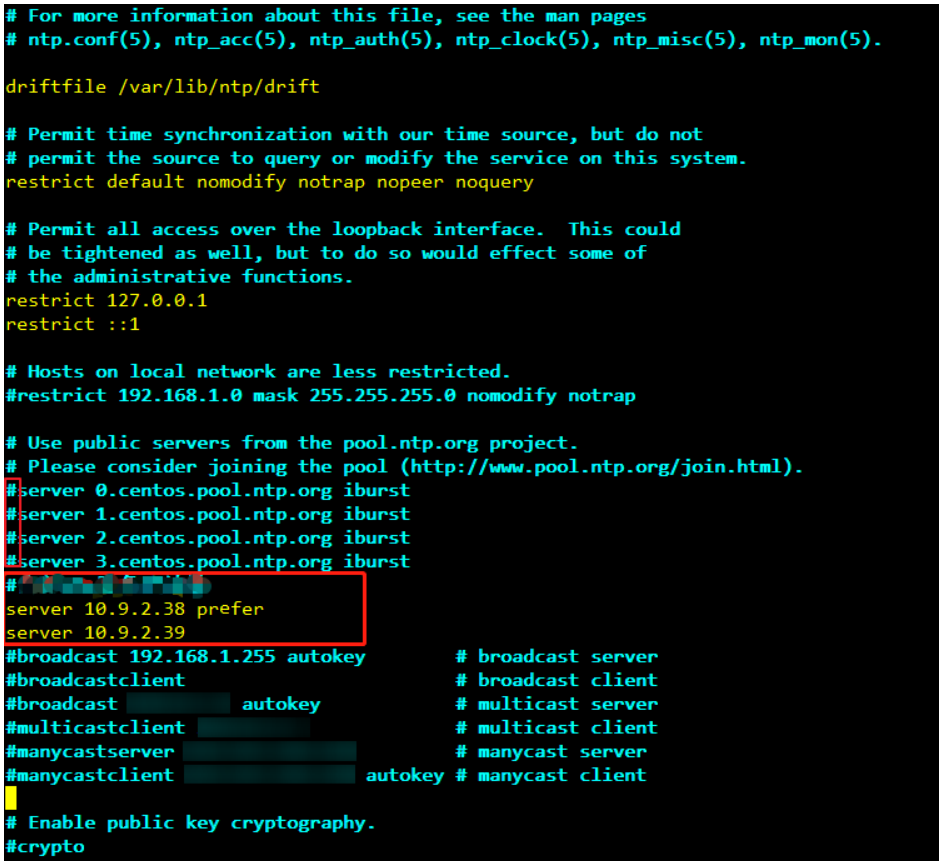
Installing a Client on a Node Outside a Cluster

1. Create an ECS that meets the requirements in **Prerequisites**.
2. Perform NTP time synchronization to synchronize the time of nodes outside the cluster with that of the MRS cluster.

a. Run the **vi /etc/ntp.conf** command to edit the NTP client configuration file, add the IP addresses of the master node in the MRS cluster, and comment out the IP address of other servers.

```
server master1_ip prefer
server master2_ip
```

Figure 6-2 Adding the master node IP addresses



b. Run the **service ntpd stop** command to stop the NTP service.

c. Run the **/usr/sbin/ntpdate** *IP address of the active master node* command to manually synchronize time.

d. Run the **service ntpd start** or **systemctl restart ntpd** command to start the NTP service.

e. Run the **ntpstat** command to check the time synchronization result.

3. Perform the following steps to download the cluster client software package from FusionInsight Manager, copy the package to the ECS node, and install the client:

a. Log in to FusionInsight Manager and download the cluster client to the specified directory on the active management node by referring to **Accessing FusionInsight Manager (MRS 3.x or Later)** and **Installing a Client on a Node Inside a Cluster**.

b. Log in to the active management node as user **root** and run the following command to copy the client installation package to the target node:

scp -p /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_Client.tar *IP address of the node where the client is to be installed*:**/tmp**

c. Log in to the node on which the client is to be installed as the client user.

Run the following commands to install the client. If the user does not have operation permissions on the client software package and client installation directory, grant the permissions using the **root** user.

cd /tmp

tar -xvf FusionInsight_Cluster_1_Services_Client.tar

tar -xvf FusionInsight_Cluster_1_Services_ClientConfig.tar

cd FusionInsight_Cluster_1_Services_ClientConfig

./install.sh /opt/client

d. Run the following commands to switch to the client directory and configure environment variables:

cd /opt/client

source bigdata_env

e. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

f. Run the client command of a component directly.

For example, run the **hdfs dfs -ls /** command to view files in the HDFS root directory.

6.1.2 Installing a Client (Versions Earlier Than 3.x)

Scenario

An MRS client is required. The MRS cluster client can be installed on the Master or Core node in the cluster or on a node outside the cluster.

After a cluster of versions earlier than MRS 3.x is created, a client is installed on the active Master node by default. You can directly use the client. The installation directory is **/opt/client**.

For details about how to install a client of MRS 3.x or later, see **Installing a Client (Version 3.x or Later)**.

NOTE

If a client has been installed on the node outside the MRS cluster and the client only needs to be updated, update the client using the user who installed the client, for example, user **root**.

Prerequisites

- An ECS has been prepared. For details about the OS and its version of the ECS, see **Table 6-2**.

Table 6-2 Reference list

OS	Supported Version
EulerOS	• Available: EulerOS 2.2 • Available: EulerOS 2.3 • Available: EulerOS 2.5

For example, a user can select an ECS running the EulerOS.

In addition, sufficient disk space is allocated for the ECS, for example, 40 GB.

- The ECS and the MRS cluster are in the same VPC.
- The security group of the ECS is the same as that of the Master node of the MRS cluster.

  If this requirement is not met, modify the ECS security group or configure the inbound and outbound rules of the ECS security group to allow the ECS security group to be accessed by all security groups of MRS cluster nodes.
- To enable users to log in to a Linux ECS using a password (SSH), see ***Instances*** > ***Logging In to a Linux ECS*** > ***Login Using an SSH Password*** *in the Elastic Cloud Server User Guide*.

Installing a Client on the Core Node

1. Log in to MRS Manager and choose **Services** > **Download Client** to download the client installation package to the active management node.

   NOTE

   If only the client configuration file needs to be updated, see method 2 in **Updating a Client (Versions Earlier Than 3.x)**.
2. Use the IP address to search for the active management node, and log in to the active management node using VNC.
3. Log in to the active management node, and run the following command to switch the user:

   sudo su - omm
4. On the MRS management console, view the IP address on the **Nodes** tab page of the specified cluster.

   Record the IP address of the Core node where the client is to be used.
5. On the active management node, run the following command to copy the client installation package to the Core node:

   scp -p /tmp/MRS-client/MRS_Services_Client.tar *IP address of the Core node*:**/opt/client**
6. Log in to the Core node as user **root**.

   Master nodes support Cloud-Init. The preset username for Cloud-Init is **root** and the password is the one you set during cluster creation.
7. Run the following commands to install the client:

   cd /opt/client

   tar -xvf MRS_Services_Client.tar

tar -xvf MRS_Services_ClientConfig.tar

cd /opt/client/MRS_Services_ClientConfig

./install.sh *Client installation directory*

For example, run the following command:

./install.sh /opt/client

8. For details about how to use the client, see **Using an MRS Client**.

Using an MRS Client

1. On the node where the client is installed, run the **sudo su - omm** command to switch the user. Run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *MRS cluster user*

   Example: **kinit admin**

   NOTE

   User **admin** is created by default for MRS clusters with Kerberos authentication enabled and is used for administrators to maintain the clusters.

4. Run the client command of a component directly.

   For example, run the **hdfs dfs -ls /** command to view files in the HDFS root directory.

Installing a Client on a Node Outside the Cluster

Step 1 Create an ECS that meets the requirements in the prerequisites.

Step 2 Log in to MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Then, choose **Services**.

Step 3 Click **Download Client**.

Step 4 In **Client Type**, select **All client files**.

Step 5 In **Download To**, select **Remote host**.

Step 6 Set **Host IP Address** to the IP address of the ECS, **Host Port** to **22**, and **Save Path** to **/tmp**.

- If the default port **22** for logging in to an ECS using SSH has been changed, set **Host Port** to the new port.
- **Save Path** contains a maximum of 256 characters.

Step 7 Set **Login User** to **root**.

If other users are used, ensure that the users have read, write, and execute permission on the save path.

Step 8 Select **Password** or **SSH Private Key** for **Login Mode**.

- **Password**: Enter the password of user **root** set during cluster creation.
- **SSH Private Key**: Select and upload the key file used for creating the cluster.

Step 9 Click **OK** to generate a client file.

If the following information is displayed, the client package is saved. Click **Close**. Obtain the client file from the save path on the remote host that is set when the client is downloaded.

```
Client files downloaded to the remote host successfully.
```

If the following information is displayed, check the username, password, and security group configurations of the remote host. Ensure that the username and password are correct and an inbound rule of the SSH (22) port has been added to the security group of the remote host. And then, go to **Step 2** to download the client again.

```
Failed to connect to the server. Please check the network connection or parameter settings.
```

NOTE

Generating a client will occupy a large number of disk I/Os. You are advised not to download a client when the cluster is being installed, started, and patched, or in other unstable states.

Step 10 Log in to the ECS using VNC. For details, see **Instance** > **Logging In to a Linux** > **Logging In to a Linux** in the *Elastic Cloud Server User Guide*

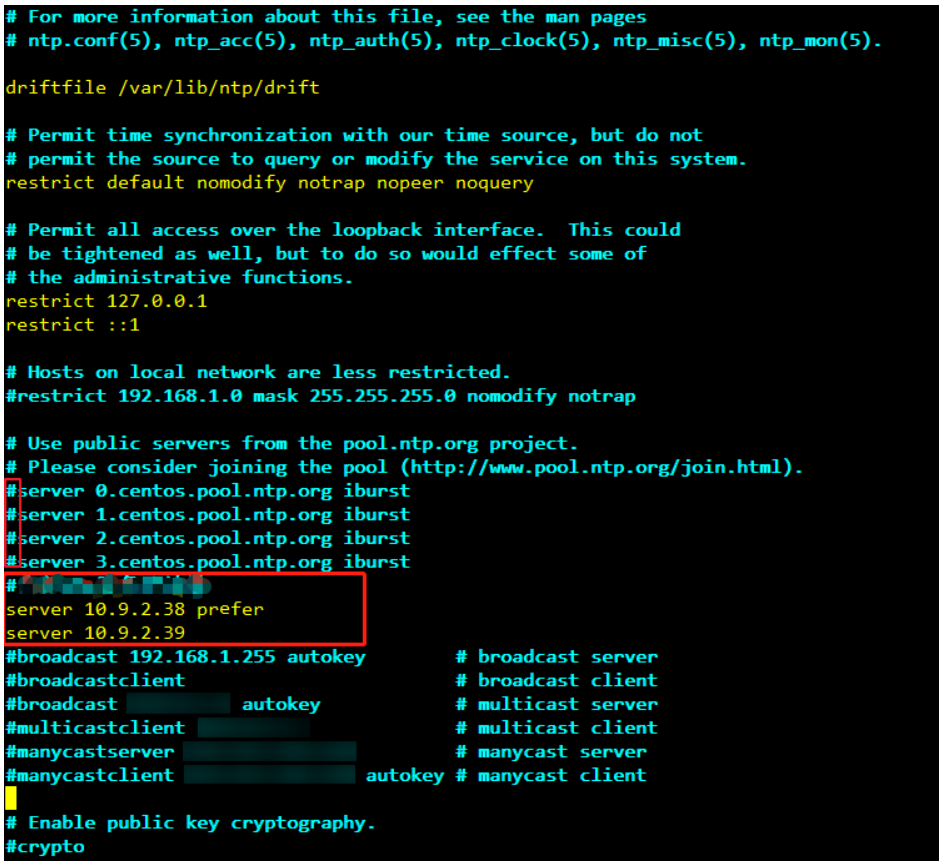
All images support Cloud-Init. The preset username for Cloud-Init is **root** and the password is the one you set during cluster creation. It is recommended that you change the password upon the first login.

Step 11 Perform NTP time synchronization to synchronize the time of nodes outside the cluster with the time of the MRS cluster.

1. Check whether the NTP service is installed. If it is not installed, run the **yum install ntp -y** command to install it.
2. Run the **vim /etc/ntp.conf** command to edit the NTP client configuration file, add the IP address of the Master node in the MRS cluster, and comment out the IP addresses of other servers.

   ```
   server master1_ip prefer
   server master2_ip
   ```

Figure 6-3 Adding the Master node IP addresses



3. Run the **service ntpd stop** command to stop the NTP service.
4. Run the**/usr/sbin/ntpdate** *IP address of the active Master node* command to manually synchronize the time.
5. Run the **service ntpd start** or **systemctl restart ntpd** command to start the NTP service.
6. Run the **ntpstat** command to check the time synchronization result:

Step 12 On the ECS, switch to user **root** and copy the installation package in **Save Path** in **Step 6** to the **/opt** directory. For example, if **Save Path** is set to **/tmp**, run the following commands:

sudo su - root

cp /tmp/MRS_Services_Client.tar /opt

Step 13 Run the following command in the **/opt** directory to decompress the package and obtain the verification file and the configuration package of the client:

tar -xvf MRS_Services_Client.tar

Step 14 Run the following command to verify the configuration file package of the client:

sha256sum -c MRS_Services_ClientConfig.tar.sha256

The command output is as follows:

```
MRS_Services_ClientConfig.tar: OK
```

Step 15 Run the following command to decompress **MRS_Services_ClientConfig.tar**:

tar -xvf MRS_Services_ClientConfig.tar

Step 16 Run the following command to install the client to a new directory, for example, **/opt/Bigdata/client**. A directory is automatically generated during the client installation.

sh /opt/MRS_Services_ClientConfig/install.sh /opt/Bigdata/client

If the following information is displayed, the client has been successfully installed:

```
Components client installation is complete.
```

Step 17 Check whether the IP address of the ECS node is connected to the IP address of the cluster Master node.

For example, run the following command: **ping** *Master node IP address*.

- If yes, go to **Step 18**.
- If no, check whether the VPC and security group are correct and whether the ECS and the MRS cluster are in the same VPC and security group, and go to **Step 18**.

Step 18 Run the following command to configure environment variables:

source /opt/Bigdata/client/bigdata_env

Step 19 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 20 Run the client command of a component.

For example, run the following command to query the HDFS directory:

hdfs dfs -ls /

----End

6.2 Updating a Client

6.2.1 Updating a Client (Version 3.x or Later)

A cluster provides a client for you to connect to a server, view task results, or manage data. If you modify service configuration parameters on Manager and restart the service, you need to download and install the client again or use the configuration file to update the client.

Updating the Client Configuration

Method 1:

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Click the name of the cluster to be operated in the **Cluster** drop-down list.

Step 2 Choose **More** > **Download Client** > **Configuration Files Only**.

The generated compressed file contains the configuration files of all services.

Step 3 Determine whether to generate a configuration file on the cluster node.

- If yes, select **Save to Path**, and click **OK** to generate the client file. By default, the client file is generated in **/tmp/FusionInsight-Client** on the active management node. You can also store the client file in other directories, and user **omm** has the read, write, and execute permissions on the directories. Then go to **Step 4**.
- If no, click **OK**, specify a local save path, and download the complete client. Wait until the download is complete and go to **Step 4**.

Step 4 Use WinSCP to save the compressed file to the client installation directory, for example, **/opt/hadoopclient**, as the client installation user.

Step 5 Decompress the software package.

Run the following commands to go to the directory where the client is installed, and decompress the file to a local directory. For example, the downloaded client file is **FusionInsight_Cluster_1_Services_Client.tar**.

cd /opt/hadoopclient

tar -xvf FusionInsight_Cluster_1_Services_Client.tar

Step 6 Verify the software package.

Run the following command to verify the decompressed file and check whether the command output is consistent with the information in the **sha256** file.

sha256sum -c FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar.sha256

```
FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar: OK
```

Step 7 Decompress the package to obtain the configuration file.

tar -xvf FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar

Step 8 Run the following command in the client installation directory to update the client using the configuration file:

sh refreshConfig.sh *Client installation directory Directory where the configuration file is located*

For example, run the following command:

sh refreshConfig.sh /opt/hadoopclient /opt/hadoopclient/FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

----End

Method 2:

Step 1 Log in to the client installation node as user **root**.

Step 2 Go to the client installation directory, for example, **/opt/hadoopclient** and run the following commands to update the configuration file:

cd /opt/hadoopclient

sh autoRefreshConfig.sh

Step 3 Enter the username and password of the FusionInsight Manager administrator and the floating IP address of FusionInsight Manager.

Step 4 Enter the names of the components whose configuration needs to be updated. Use commas (,) to separate the component names. Press **Enter** to update the configurations of all components if necessary.

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

----End

6.2.2 Updating a Client (Versions Earlier Than 3.x)

NOTE

This section applies to clusters of versions earlier than MRS 3.x. For MRS 3.x or later, see **Updating a Client (Version 3.x or Later)**.

Updating a Client Configuration File

Scenario

An MRS cluster provides a client for you to connect to a server, view task results, or manage data. Before using an MRS client, you need to download and update the client configuration file if service configuration parameters are modified and a service is restarted or the service is merely restarted on MRS Manager.

During cluster creation, the original client is stored in the **/opt/client** directory on all nodes in the cluster by default. After the cluster is created, only the client of a Master node can be directly used. To use the client of a Core node, you need to update the client configuration file first.

Procedure

Method 1:

Step 1 Log in to MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Then, choose **Services**.

Step 2 Click **Download Client**.

Set **Client Type** to **Only configuration files**, **Download To** to **Server**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default. You can customize the file path.

Step 3 Query and log in to the active Master node.

Step 4 If you use the client in the cluster, run the following command to switch to user **omm**. If you use the client outside the cluster, switch to user **root**.

sudo su - omm

Step 5 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 6 Run the following command to update client configurations:

sh refreshConfig.sh *Client installation directory Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/Bigdata/client /tmp/MRS-client/MRS_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully.

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

----End

Method 2:

Step 1 After the cluster is installed, run the following command to switch to user **omm**. If you use the client outside the cluster, switch to user **root**.

sudo su - omm

Step 2 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 3 Run the following command and enter the name of an MRS Manager user with the download permission and its password (for example, the username is **admin** and the password is the one set during cluster creation) as prompted to update client configurations.

sh autoRefreshConfig.sh

Step 4 After the command is executed, the following information is displayed, where *XXX* indicates the name of the component installed in the cluster. To update client configurations of all components, press **Enter**. To update client configurations of some components, enter the component names and separate them with commas (,).

```
Components "xxx" have been installed in the cluster. Please input the comma-separated names of the components for which you want to update client configurations. If you press Enter without inputting any component name, the client configurations of all components will be updated:
```

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

If the following information is displayed, the username or password is incorrect.

```
login manager failed,Incorrect username or password.
```

NOTE

- This script automatically connects to the cluster and invokes the **refreshConfig.sh** script to download and update the client configuration file.
- By default, the client uses the floating IP address specified by **wsom=xxx** in the **Version** file in the installation directory to update the client configurations. To update the configuration file of another cluster, modify the value of **wsom=xxx** in the **Version** file to the floating IP address of the corresponding cluster before performing this step.

----End

Fully Updating the Original Client of the Active Master Node

Scenario

During cluster creation, the original client is stored in the **/opt/client** directory on all nodes in the cluster by default. The following uses **/opt/Bigdata/client** as an example.

- For a normal MRS cluster, you will use the pre-installed client on a Master node to submit a job on the management console page.
- You can also use the pre-installed client on the Master node to connect to a server, view task results, and manage data.

After installing the patch on the cluster, you need to update the client on the Master node to ensure that the functions of the built-in client are available.

Procedure

Step 1 Log in to MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Then, choose **Services**.

Step 2 Click **Download Client**.

Set **Client Type** to **All client files**, **Download To** to **Server**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default. You can customize the file path.

Step 3 Query and log in to the active Master node.

Step 4 On the ECS, switch to user **root** and copy the installation package to the **/opt** directory.

sudo su - root

cp /tmp/MRS-client/MRS_Services_Client.tar /opt

Step 5 Run the following command in the **/opt** directory to decompress the package and obtain the verification file and the configuration package of the client:

tar -xvf MRS_Services_Client.tar

Step 6 Run the following command to verify the configuration file package of the client:

sha256sum -c MRS_Services_ClientConfig.tar.sha256

The command output is as follows:

```
MRS_Services_ClientConfig.tar: OK
```

Step 7 Run the following command to decompress **MRS_Services_ClientConfig.tar**:

tar -xvf MRS_Services_ClientConfig.tar

Step 8 Run the following command to move the original client to the **/opt/Bigdata/client_bak** directory:

mv /opt/Bigdata/client /opt/Bigdata/client_bak

Step 9 Run the following command to install the client in a new directory. The client path must be **/opt/Bigdata/client**.

sh /opt/MRS_Services_ClientConfig/install.sh /opt/Bigdata/client

If the following information is displayed, the client has been successfully installed:

```
Components client installation is complete.
```

Step 10 Run the following command to modify the user and user group of the **/opt/Bigdata/client** directory:

chown omm:wheel /opt/Bigdata/client -R

Step 11 Run the following command to configure environment variables:

source /opt/Bigdata/client/bigdata_env

Step 12 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 13 Run the client command of a component.

For example, run the following command to query the HDFS directory:

hdfs dfs -ls /

----End

Fully Updating the Original Client of the Standby Master Node

Step 1 Repeat **Step 1** to **Step 3** to log in to the standby Master node, and run the following command to switch to user **omm**:

sudo su - omm

Step 2 Run the following command on the standby master node to copy the downloaded client package from the active master node:

scp omm@*master1 nodeIP address***:/tmp/MRS-client/MRS_Services_Client.tar /tmp/MRS-client/**

NOTE

- In this command, **master1** node is the active master node.
- **/tmp/MRS-client/** is an example target directory of the standby master node.

Step 3 Repeat **Step 4** to **Step 13** to update the client of the standby Master node.

----End

6.3 Using the Client of Each Component

6.3.1 Using a ClickHouse Client

ClickHouse is a column-based database oriented to online analysis and processing. It supports SQL query and provides good query performance. The aggregation analysis and query performance based on large and wide tables is excellent, which is one order of magnitude faster than other analytical databases.

Prerequisites

You have installed the client, for example, in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory. Before using the client, download and update the client configuration file, and ensure that the active management node of Manager is available.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create ClickHouse tables. For details about how to bind a role to the user, see **ClickHouse User and Permission Management**. If Kerberos authentication is disabled for the current cluster, skip this step.

1. Run the following command if it is an MRS 3.1.0 cluster:

   export CLICKHOUSE_SECURITY_ENABLED=true

2. **kinit** *Component service user*

   Example: **kinit clickhouseuser**

Step 5 Run the client command of the ClickHouse component.

Run the **clickhouse -h** command to view the command help of ClickHouse.

The command output is as follows:

```
Use one of the following commands:
clickhouse local [args]
clickhouse client [args]
clickhouse benchmark [args]
clickhouse server [args]
clickhouse performance-test [args]
clickhouse extract-from-config [args]
clickhouse compressor [args]
clickhouse format [args]
clickhouse copier [args]
clickhouse obfuscator [args]
...
```

Run the **clickhouse client** command to connect to the ClickHouse serverif MRS 3.1.0 or later.

- Using SSL for login when Kerberos authentication is disabled for the current cluster:

  clickhouse client --host *IP address of the ClickHouse instance* **--user** *Username* **--password** *Password* **--port** 9440 **--secure**

- Using SSL for login when Kerberos authentication is enabled for the current cluster:

  You must create a user on Manager because there is no default user. For details, see **ClickHouse User and Permission Management**.

  After the user authentication is successful, you do not need to carry the **--user** and **--password** parameters when logging in to the client as the authenticated user.

  clickhouse client --host *IP address of the ClickHouse instance* **--port** 9440 **--secure**

  The following table describes the parameters of the **clickhouse client** command.

Table 6-3 Parameters of the **clickhouse client** command

Parameter	Description
--host	Host name of the server. The default value is **localhost**. You can use the host name or IP address of the node where the ClickHouse instance is located. **NOTE** You can log in to FusionInsight Manager and choose **Cluster** > **Services** > **ClickHouse** > **Instance** to obtain the service IP address of the ClickHouseServer instance.
--port	Port for connection. • If the SSL security connection is used, the default port number is **9440**, the parameter **--secure** must be carried. For details about the port number, search for the **tcp_port_secure** parameter in the ClickHouseServer instance configuration. • If non-SSL security connection is used, the default port number is **9000**, the parameter **--secure** does not need to be carried. For details about the port number, search for the **tcp_port** parameter in the ClickHouseServer instance configuration.

Parameter	Description
--user	Username. You can create the user on Manager and bind a role to the user. For details, see **ClickHouse User and Permission Management**. • If Kerberos authentication is enabled for the current cluster and the user authentication is successful, you do not need to carry the **--user** and **--password** parameters when logging in to the client as the authenticated user. You must create a user with this name on Manager because there is no default user in the Kerberos cluster scenario. • If Kerberos authentication is not enabled for the current cluster, you can specify a user and its password created on Manager when logging in to the client. If the user and password parameters are not carried, user **default** is used for login by default.
--password	Password. The default password is an empty string. This parameter is used together with the **--user** parameter. You can set a password when creating a user on Manager.
--query	Query to process when using non-interactive mode.
--database	Current default database. The default value is **default**, which is the default configuration on the server.
--multiline	If this parameter is specified, multiline queries are allowed. (**Enter** only indicates line feed and does not indicate that the query statement is complete.)
--multiquery	If this parameter is specified, multiple queries separated with semicolons (;) can be processed. This parameter is valid only in non-interactive mode.
--format	Specified default format used to output the result.
--vertical	If this parameter is specified, the result is output in vertical format by default. In this format, each value is printed on a separate line, which helps to display a wide table.
--time	If this parameter is specified, the query execution time is printed to **stderr** in non-interactive mode.
--stacktrace	If this parameter is specified, stack trace information will be printed when an exception occurs.
--config-file	Name of the configuration file.
--secure	If this parameter is specified, the server will be connected in SSL mode.
--history_file	Path of files that record command history.

Parameter	Description
--param_<name>	Query with parameters. Pass values from the client to the server. For details, see **https://clickhouse.tech/docs/en/interfaces/cli/#cli-queries-with-parameters**.

----End

6.3.2 Using a Flink Client

This section describes how to use Flink to run wordcount jobs.

Prerequisites

- Flink has been installed in an MRS cluster.
- The cluster runs properly and the client has been correctly installed, for example, in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Using the Flink Client (Versions Earlier Than MRS 3.x)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to initialize environment variables:

source /opt/hadoopclient/bigdata_env

Step 4 If Kerberos authentication is enabled for the cluster, perform the following steps. If not, skip this whole step.

1. Prepare a user for submitting Flink jobs..
2. Log in to Manager and download the authentication credential.

   Log in to Manager of the cluster. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Choose **System Settings** > **User Management**. In the **Operation** column of the row that contains the added user, choose **More** > **Download Authentication Credential**.
3. Decompress the downloaded authentication credential package and copy the **user.keytab** file to the client node, for example, to the **/opt/hadoopclient/Flink/flink/conf** directory on the client node. If the client is installed on a node outside the cluster, copy the **krb5.conf** file to the **/etc/** directory on this node.
4. Configure security authentication by adding the **keytab** path and username in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** configuration file.

   security.kerberos.login.keytab: *<user.keytab file path>*

   security.kerberos.login.principal: *<Username>*

   Example:

   security.kerberos.login.keytab: /opt/hadoopclient/Flink/flink/conf/user.keytab

security.kerberos.login.principal: test

5. Generate the **generate_keystore.sh** script by referring to "Using Flink" > "Reference" > "Example of Issuing a Certificate" in *MapReduce Service Component Operation Guide* and place it in the **bin** directory of the Flink client.In the **bin** directory of the Flink client, run the following command to perform security hardening and set password to a new one for submitting jobs. For details, see "Using Flink" > "Security Hardening" > "Authentication and Encryption" in *MapReduce Service Component Operation Guide*.

   sh generate_keystore.sh <*password*>

   The script automatically replaces the SSL value in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** file. For an MRS 2.*x* or earlier security cluster, external SSL is disabled by default. To enable external SSL, configure the parameter and run the script again. For details, see "Using Flink" > "Security Hardening" in *MapReduce Service Component Operation Guide*.

   NOTE

   - You do not need to manually generate the **generate_keystore.sh** script.
   - After authentication and encryption, the generated **flink.keystore**, **flink.truststore**, and **security.cookie** items are automatically filled in the corresponding configuration items in **flink-conf.yaml**.

6. Configure paths for the client to access the **flink.keystore** and **flink.truststore** files.
   - Absolute path: After the script is executed, the file path of **flink.keystore** and **flink.truststore** is automatically set to the absolute path **/opt/hadoopclient/Flink/flink/conf/** in the **flink-conf.yaml** file. In this case, you need to move the **flink.keystore** and **flink.truststore** files from the **conf** directory to this absolute path on the Flink client and Yarn nodes.
   - Relative path: Perform the following steps to set the file path of **flink.keystore** and **flink.truststore** to the relative path and ensure that the directory where the Flink client command is executed can directly access the relative paths.
     i. Create a directory, for example, **ssl**, in **/opt/hadoopclient/Flink/flink/conf/**.

        cd /opt/hadoopclient/Flink/flink/conf/

        mkdir ssl

     ii. Move the **flink.keystore** and **flink.truststore** files to the **/opt/hadoopclient/Flink/flink/conf/ssl/** directory.

        mv flink.keystore ssl/

        mv flink.truststore ssl/

     iii. Change the values of the following parameters to relative paths in the **flink-conf.yaml** file:

```
security.ssl.internal.keystore: ssl/flink.keystore
security.ssl.internal.truststore: ssl/flink.truststore
```

Step 5 Run a wordcount job.

NOTICE

To submit or run jobs on Flink, the user must have the following permissions:

- If Ranger authentication is enabled, the current user must belong to the **hadoop** group or the user has been granted the **/flink** read and write permissions in Ranger.
- If Ranger authentication is disabled, the current user must belong to the **hadoop** group.

- Normal cluster (Kerberos authentication disabled)
  - Run the following commands to start a session and submit a job in the session:

    yarn-session.sh -nm "*session-name***"**

    flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - Run the following command to submit a single job on Yarn:

    flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
- Security cluster (Kerberos authentication enabled)
  - If the **flink.keystore** and **flink.truststore** file are stored in the absolute path:
    - Run the following commands to start a session and submit a job in the session:

      yarn-session.sh -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
    - Run the following command to submit a single job on Yarn:

      flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - If the **flink.keystore** and **flink.truststore** files are stored in the relative path:
    - In the same directory of SSL, run the following commands to start a session and submit jobs in the session. The SSL directory is a relative path. For example, if the SSL directory is **opt/hadoopclient/Flink/flink/conf/**, then run the following commands in this directory:

      yarn-session.sh -t ssl/ -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
    - Run the following command to submit a single job on Yarn:

      flink run -m yarn-cluster -yt ssl/ /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar

Step 6 After the job has been successfully submitted, the following information is displayed on the client:

Figure 6-4 Job submitted successfully on Yarn

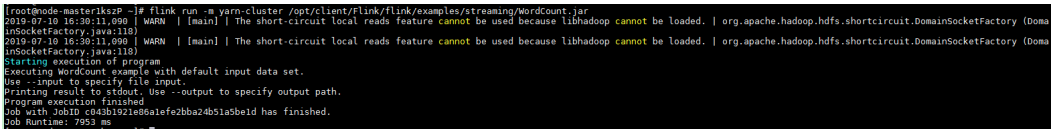
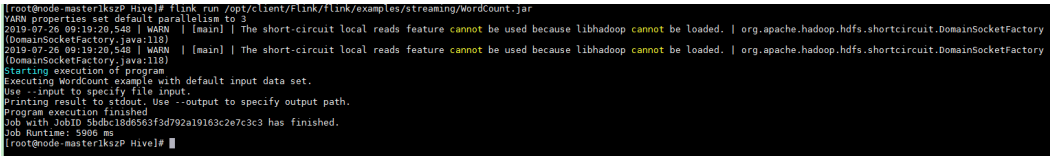


Figure 6-5 Session started successfully

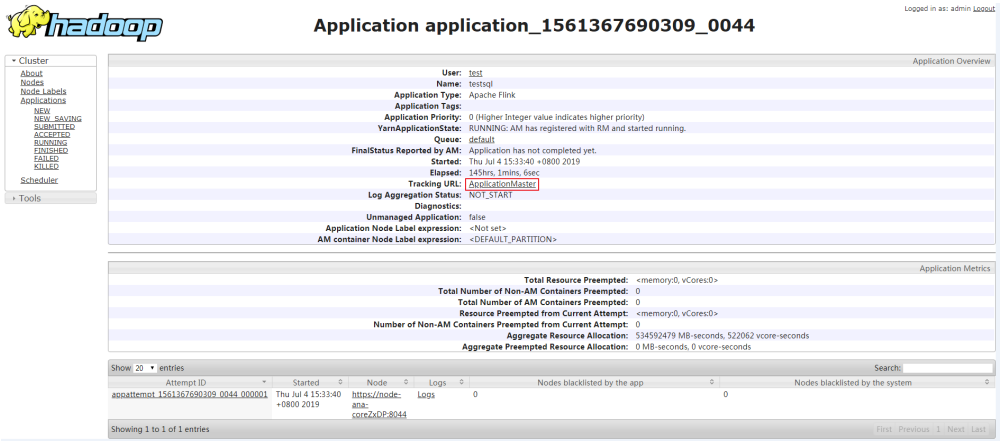
Figure 6-6 Job submitted successfully in the session



Step 7 Go to the native YARN service page, find the application of the job, and click the application name to go to the job details page. For details, see "Using Flink" > "Viewing Flink Job Information" in *MapReduce Service Component Operation Guide*.

- If the job is not completed, click **Tracking URL** to go to the native Flink page and view the job running information.
- If the job submitted in a session has been completed, you can click **Tracking URL** to log in to the native Flink service page to view job information.

Figure 6-7 Application



----End

Using the Flink Client (MRS 3.x or Later)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to initialize environment variables:

source /opt/hadoopclient/bigdata_env

Step 4 If Kerberos authentication is enabled for the cluster, perform the following steps. If not, skip this whole step.

1. Prepare a user for submitting Flink jobs.
2. Log in to Manager and download the authentication credential.

   Log in to Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Choose **System** > **Permission** > **Manage User**. On the displayed page, locate the row that contains the added user, click **More** in the **Operation** column, and select **Download authentication credential**.
3. Decompress the downloaded authentication credential package and copy the **user.keytab** file to the client node, for example, to the **/opt/hadoopclient/Flink/flink/conf** directory on the client node. If the client is installed on a node outside the cluster, copy the **krb5.conf** file to the **/etc/** directory on this node.
4. Append the service IP address of the node where the client is installed, floating IP address of Manager, and IP address of the master node to the **jobmanager.web.access-control-allow-origin** and **jobmanager.web.allow-access-address** configuration item in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** file. Use commas (,) to separate IP addresses.

   ```
   jobmanager.web.access-control-allow-origin: xx.xx.xxx.xxx,xx.xx.xxx.xxx,xx.xx.xxx.xxx
   jobmanager.web.allow-access-address: xx.xx.xxx.xxx,xx.xx.xxx.xxx,xx.xx.xxx.xxx
   ```

   NOTE

   - To obtain the service IP address of the node where the client is installed, perform the following operations:
     - Node inside the cluster:

       In the navigation tree of the MRS management console, choose **Clusters** > **Active Clusters**, select a cluster, and click its name to switch to the cluster details page.

       On the **Nodes** tab page, view the IP address of the node where the client is installed.
     - Node outside the cluster: IP address of the ECS where the client is installed.
   - To obtain the floating IP address of Manager, perform the following operations:
     - In the navigation tree of the MRS management console, choose **Clusters** > **Active Clusters**, select a cluster, and click its name to switch to the cluster details page.

       On the **Nodes** tab page, view the **Name**. The node that contains **master1** in its name is the Master1 node. The node that contains **master2** in its name is the Master2 node.
     - Log in to the Master2 node remotely, and run the **ifconfig** command. In the command output, **eth0:wsom** indicates the floating IP address of MRS Manager. Record the value of **inet**. If the floating IP address of MRS Manager cannot be queried on the Master2 node, switch to the Master1 node to query and record the floating IP address. If there is only one Master node, query and record the cluster manager IP address of the Master node.
5. Configure security authentication by adding the **keytab** path and username in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** configuration file.

security.kerberos.login.keytab: *<user.keytab file path>*

security.kerberos.login.principal: *<Username>*

Example:

security.kerberos.login.keytab: /opt/hadoopclient/Flink/flink/conf/user.keytab

security.kerberos.login.principal: test

6. Generate the **generate_keystore.sh** script by referring to "Using Flink" > "Reference" > "Example of Issuing a Certificate" in *MapReduce Service Component Operation Guide* and place it in the **bin** directory of the Flink client.In the **bin** directory of the Flink client, run the following command to perform security hardening and set password to a new one for submitting jobs. For details, see "Using Flink" > "Security Hardening" > "Authentication and Encryption" in *MapReduce Service Component Operation Guide*.

   sh generate_keystore.sh <*password*>

   The script automatically replaces the SSL value in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** file.

   sh generate_keystore.sh <*password*>

   NOTE

   After authentication and encryption, the **flink.keystore** and **flink.truststore** files are generated in the **conf** directory on the Flink client and the following configuration items are set to the default values in the **flink-conf.yaml** file:

   - Set **security.ssl.keystore** to the absolute path of the **flink.keystore** file.
   - Set **security.ssl.truststore** to the absolute path of the **flink.truststore** file.
   - Set **security.cookie** to a random password automatically generated by the **generate_keystore.sh** script.
   - By default, **security.ssl.encrypt.enabled** is set to **false** in the **flink-conf.yaml** file by default. The **generate_keystore.sh** script sets **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password** to the password entered when the **generate_keystore.sh** script is called.
   - For MRS 3.1.0 or later, if ciphertext is required and **security.ssl.encrypt.enabled** is set to **true** in the **flink-conf.yaml** file, the **generate_keystore.sh** script does not set **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password**. To obtain the values, use the Manager plaintext encryption API by running the following command: **curl -k -i -u** *Username*:*Password* **-X POST -HContent-type:application/json -d '{"plainText":"***Password***"}' 'https://***x.x.x.x***:28443/web/api/v2/tools/encrypt'**

     In the preceding command, *Username*:*Password* indicates the user name and password for logging in to the system. The password of **"plainText"** indicates the one used to call the **generate_keystore.sh** script. *x.x.x.x* indicates the floating IP address of Manager.

7. Configure paths for the client to access the **flink.keystore** and **flink.truststore** files.
   - Absolute path: After the script is executed, the file path of **flink.keystore** and **flink.truststore** is automatically set to the absolute path **/opt/hadoopclient/Flink/flink/conf/** in the **flink-conf.yaml** file. In this case, you need to move the **flink.keystore** and **flink.truststore** files from the **conf** directory to this absolute path on the Flink client and Yarn nodes.
   - Relative path: Perform the following steps to set the file path of **flink.keystore** and **flink.truststore** to the relative path and ensure that the directory where the Flink client command is executed can directly access the relative paths.

i. Create a directory, for example, **ssl**, in **/opt/hadoopclient/Flink/flink/conf/**.

cd /opt/hadoopclient/Flink/flink/conf/

mkdir ssl

ii. Move the **flink.keystore** and **flink.truststore** files to the **/opt/hadoopclient/Flink/flink/conf/ssl/** directory.

mv flink.keystore ssl/

mv flink.truststore ssl/

iii. Change the values of the following parameters to relative paths in the **flink-conf.yaml** file:

```
security.ssl.keystore: ssl/flink.keystore
security.ssl.truststore: ssl/flink.truststore
```

Step 5 Run a wordcount job.

NOTICE

To submit or run jobs on Flink, the user must have the following permissions:

- If Ranger authentication is enabled, the current user must belong to the **hadoop** group or the user has been granted the **/flink** read and write permissions in Ranger.
- If Ranger authentication is disabled, the current user must belong to the **hadoop** group.

- Normal cluster (Kerberos authentication disabled)
  - Run the following commands to start a session and submit a job in the session:

    yarn-session.sh -nm "*session-name***"**

    flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - Run the following command to submit a single job on Yarn:

    flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
- Security cluster (Kerberos authentication enabled)
  - If the **flink.keystore** and **flink.truststore** files are stored in the absolute path:
    - Run the following commands to start a session and submit a job in the session:

      yarn-session.sh -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
    - Run the following command to submit a single job on Yarn:

      flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - If the **flink.keystore** and **flink.truststore** file are stored in the relative path:

- In the same directory of SSL, run the following commands to start a session and submit jobs in the session. The SSL directory is a relative path. For example, if the SSL directory is **opt/hadoopclient/Flink/flink/conf/**, then run the following commands in this directory:

  yarn-session.sh -t ssl/ -nm "*session-name***"**

  flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar

- Run the following command to submit a single job on Yarn:

  flink run -m yarn-cluster -yt ssl/ /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar

Step 6 After the job has been successfully submitted, the following information is displayed on the client:

Figure 6-8 Job submitted successfully on Yarn

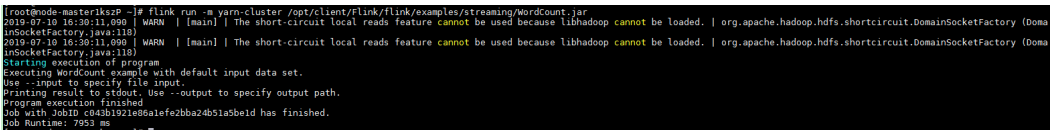


Figure 6-9 Session started successfully

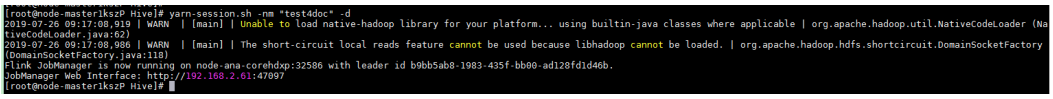
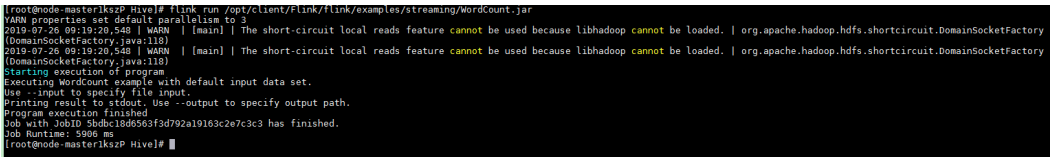


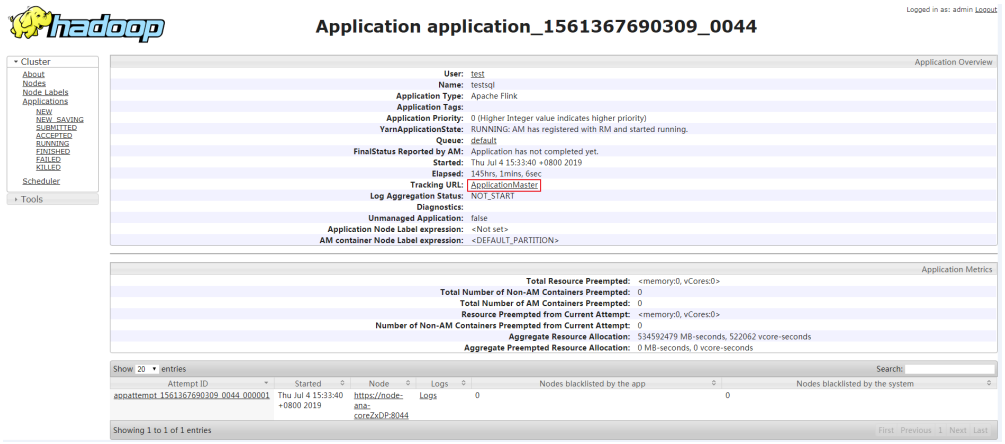
Figure 6-10 Job submitted successfully in the session



Step 7 Go to the native YARN service page, find the application of the job, and click the application name to go to the job details page. For details, see "Using Flink" > "Viewing Flink Job Information" in *MapReduce Service Component Operation Guide*.

- If the job is not completed, click **Tracking URL** to go to the native Flink page and view the job running information.
- If the job submitted in a session has been completed, you can click **Tracking URL** to log in to the native Flink service page to view job information.

Figure 6-11 Application



----End

6.3.3 Using a Flume Client

Scenario

You can use Flume to import collected log information to Kafka.

Prerequisites

- A streaming cluster with Kerberos authentication enabled has been created.
- The Flume client has been installed in a directory, for example, **/opt/Flumeclient**, on the node where logs are generated. For details about how to install the Flume client, see "Using Flume" > "Installing the Flume Client" in *MapReduce Service Component Operation Guide*. The client directory in the following operations is only an example. Change it to the actual installation directory.
- The streaming cluster can properly communicate with the node where logs are generated.

Using the Flume Client (Versions Earlier Than MRS 3.x)

NOTE

You do not need to perform **Step 2** to **Step 6** for a normal cluster.

Step 1 Install the client.

Step 2 Copy the configuration file of the authentication server from the Master1 node to the *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf** directory on the node where the Flume client resides.

The full file path is **${BIGDATA_HOME}/MRS_Current/1_X_KerberosClient/etc/kdc.conf**.

In the preceding paths, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 3 Check the service IP address of any node where the Flume role is deployed.

Log in to the cluster details page, choose *Name of the desired cluster* > **Components** > **Flume** > **Instances**, and check the service IP address of any node where the Flume role is deployed.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 4 Copy the user authentication file from this node to the *Flume client installation directory*/**fusioninsight-flume-Flume component version number/conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/MRS_XXX/install/FusionInsight-Flume-Flume component version number/flume/conf/flume.keytab**.

In the preceding paths, **XXX** indicates the product version number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 5 Copy the **jaas.conf** file from this node to the **conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/MRS_Current/1_X_Flume/etc/jaas.conf**.

In the preceding path, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 6 Log in to the Flume client node and go to the client installation directory. Run the following command to modify the file:

vi conf/jaas.conf

Change the full path of the user authentication file defined by **keyTab** to the **Flume client installation directory/fusioninsight-flume-*Flume component version number*/conf** saved in **Step 4**, and save the modification and exit.

Step 7 Run the following command to modify the **flume-env.sh** configuration file of the Flume client:

vi *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf/flume-env.sh**

Add the following information after **-XX:+UseCMSCompactAtFullCollection**:

```
-Djava.security.krb5.conf=Flume client installation directory/fusioninsight-flume-1.9.0/conf/kdc.conf -Djava.security.auth.login.config=Flume client installation directory/fusioninsight-flume-1.9.0/conf/jaas.conf -Dzookeeper.request.timeout=120000
```

For example, **"-XX:+UseCMSCompactAtFullCollection -Djava.security.krb5.conf=*Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/kdc.conf -Djava.security.auth.login.config=***Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/jaas.conf -Dzookeeper.request.timeout=120000"**

Change *Flume client installation directory* to the actual installation directory. Then save and exit.

Step 8 Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following command to restart the Flume client:

cd /opt/FlumeClient/fusioninsight-flume-*Flume component version number***/bin**

./flume-manage.sh restart

Step 9 Run the following command to modify the **properties.properties** configuration file of the Flume client:

vi *Flume client installation directory***/fusioninsight-flume-***Flume component version number***/conf/properties.properties**

Add the following information to the file:

```
################################################################################
################
client.sources = static_log_source
client.channels = static_log_channel
client.sinks = kafka_sink
################################################################################
################
#LOG_TO_HDFS_ONLINE_1

client.sources.static_log_source.type = spooldir
client.sources.static_log_source.spoolDir = PATH
client.sources.static_log_source.fileSuffix = .COMPLETED
client.sources.static_log_source.ignorePattern = ^$
client.sources.static_log_source.trackerDir = PATH
client.sources.static_log_source.maxBlobLength = 16384
client.sources.static_log_source.batchSize = 51200
client.sources.static_log_source.inputCharset = UTF-8
client.sources.static_log_source.deserializer = LINE
client.sources.static_log_source.selector.type = replicating
client.sources.static_log_source.fileHeaderKey = file
client.sources.static_log_source.fileHeader = false
client.sources.static_log_source.basenameHeader = true
client.sources.static_log_source.basenameHeaderKey = basename
client.sources.static_log_source.deletePolicy = never

client.channels.static_log_channel.type = file
client.channels.static_log_channel.dataDirs = PATH
client.channels.static_log_channel.checkpointDir = PATH
client.channels.static_log_channel.maxFileSize = 2146435071
client.channels.static_log_channel.capacity = 1000000
client.channels.static_log_channel.transactionCapacity = 612000
client.channels.static_log_channel.minimumRequiredSpace = 524288000

client.sinks.kafka_sink.type = org.apache.flume.sink.kafka.KafkaSink
client.sinks.kafka_sink.kafka.topic = flume_test
client.sinks.kafka_sink.kafka.bootstrap.servers = XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka
port number,XXX.XXX.XXX.XXX:Kafka port number
client.sinks.kafka_sink.flumeBatchSize = 1000
client.sinks.kafka_sink.kafka.producer.type = sync
client.sinks.kafka_sink.kafka.security.protocol  = SASL_PLAINTEXT
client.sinks.kafka_sink.kafka.kerberos.domain.name = hadoop.XXX.com
client.sinks.kafka_sink.requiredAcks = 0

client.sources.static_log_source.channels = static_log_channel
client.sinks.kafka_sink.channel = static_log_channel
```

Modify the following parameters as required. Then save and exit the file.

- spoolDir

- trackerDir
- dataDirs
- checkpointDir
- topic

  If the topic does not exist in Kafka, the topic is automatically created by default.
- kafka.bootstrap.servers

  By default, the port for a security cluster is port 21007 and that for a normal cluster is port 9092.
- kafka.security.protocol

  Set this parameter to **SASL_PLAINTEXT** for a security cluster and **PLAINTEXT** for a normal cluster.
- **kafka.kerberos.domain.name**

  You do not need to set this parameter for a normal cluster. For a security cluster, the value of this parameter is the value of **kerberos.domain.name** in the Kafka cluster.

  You can check **${BIGDATA_HOME}/MRS_Current/1_X_Broker/etc/server.properties** on the node where the broker instance resides.

  In the preceding paths, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 10 The Flume client automatically loads the information in the **properties.properties** file.

After new log files are generated in the directory specified by **spoolDir**, the logs will be sent to Kafka producers and can be consumed by Kafka consumers.

----End

Using the Flume Client (MRS 3.x or Later)

NOTE

You do not need to perform **Step 2** to **Step 6** for a normal cluster.

Step 1 Install the client.

Step 2 Copy the configuration file of the authentication server from the Master1 node to the *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf** directory on the node where the Flume client resides.

The full file path is **${BIGDATA_HOME}/FusionInsight_Current/1_X_KerberosClient/etc/kdc.conf**. In the preceding path, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 3 Check the service IP address of any node where the Flume role is deployed.

Log in to FusionInsight Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Choose **Cluster > Services > Flume > Instance**. Check the service IP address of any node where the Flume role is deployed.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 4 Copy the user authentication file from this node to the *Flume client installation directory*/**fusioninsight-flume-Flume component version number/conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/FusionInsight_Porter_XXX/install/FusionInsight-Flume-Flume component version number/flume/conf/flume.keytab**.

In the preceding paths, **XXX** indicates the product version number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 5 Copy the **jaas.conf** file from this node to the **conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/FusionInsight_Current/1_X_Flume/etc/jaas.conf**.

In the preceding path, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 6 Log in to the Flume client node and go to the client installation directory. Run the following command to modify the file:

vi conf/jaas.conf

Change the full path of the user authentication file defined by **keyTab** to the **Flume client installation directory/fusioninsight-flume-*Flume component version number*/conf** saved in **Step 4**, and save the modification and exit.

Step 7 Run the following command to modify the **flume-env.sh** configuration file of the Flume client:

vi *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf/flume-env.sh**

Add the following information after **-XX:+UseCMSCompactAtFullCollection**:

```
-Djava.security.krb5.conf=Flume client installation directory/fusioninsight-flume-1.9.0/conf/kdc.conf -Djava.security.auth.login.config=Flume client installation directory/fusioninsight-flume-1.9.0/conf/jaas.conf -Dzookeeper.request.timeout=120000
```

For example, **"-XX:+UseCMSCompactAtFullCollection -Djava.security.krb5.conf=*Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/kdc.conf -Djava.security.auth.login.config=***Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/jaas.conf -Dzookeeper.request.timeout=120000"**

Change *Flume client installation directory* to the actual installation directory. Then save and exit.

Step 8 Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following command to restart the Flume client:

cd /opt/FlumeClient/fusioninsight-flume-*Flume component version number***/bin**

./flume-manage.sh restart

Step 9 Run the following command to modify the **properties.properties** configuration file of the Flume client:

vi *Flume client installation directory***/fusioninsight-flume-***Flume component version number***/conf/properties.properties**

Add the following information to the file:

```
################################################################################################
client.sources = static_log_source
client.channels = static_log_channel
client.sinks = kafka_sink
################################################################################################
#LOG_TO_HDFS_ONLINE_1

client.sources.static_log_source.type = spooldir
client.sources.static_log_source.spoolDir = PATH
client.sources.static_log_source.fileSuffix = .COMPLETED
client.sources.static_log_source.ignorePattern = ^$
client.sources.static_log_source.trackerDir = PATH
client.sources.static_log_source.maxBlobLength = 16384
client.sources.static_log_source.batchSize = 51200
client.sources.static_log_source.inputCharset = UTF-8
client.sources.static_log_source.deserializer = LINE
client.sources.static_log_source.selector.type = replicating
client.sources.static_log_source.fileHeaderKey = file
client.sources.static_log_source.fileHeader = false
client.sources.static_log_source.basenameHeader = true
client.sources.static_log_source.basenameHeaderKey = basename
client.sources.static_log_source.deletePolicy = never

client.channels.static_log_channel.type = file
client.channels.static_log_channel.dataDirs = PATH
client.channels.static_log_channel.checkpointDir = PATH
client.channels.static_log_channel.maxFileSize = 2146435071
client.channels.static_log_channel.capacity = 1000000
client.channels.static_log_channel.transactionCapacity = 612000
client.channels.static_log_channel.minimumRequiredSpace = 524288000

client.sinks.kafka_sink.type = org.apache.flume.sink.kafka.KafkaSink
client.sinks.kafka_sink.kafka.topic = flume_test
client.sinks.kafka_sink.kafka.bootstrap.servers = XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka port number
client.sinks.kafka_sink.flumeBatchSize = 1000
client.sinks.kafka_sink.kafka.producer.type = sync
client.sinks.kafka_sink.kafka.security.protocol  = SASL_PLAINTEXT
client.sinks.kafka_sink.kafka.kerberos.domain.name = hadoop.XXX.com
client.sinks.kafka_sink.requiredAcks = 0

client.sources.static_log_source.channels = static_log_channel
client.sinks.kafka_sink.channel = static_log_channel
```

Modify the following parameters as required. Then save and exit the file.

- spoolDir
- trackerDir
- dataDirs
- checkpointDir
- topic

If the topic does not exist in Kafka, the topic is automatically created by default.

- kafka.bootstrap.servers

  By default, the port for a security cluster is port 21007 and that for a normal cluster is port 9092.

- kafka.security.protocol

  Set this parameter to **SASL_PLAINTEXT** for a security cluster and **PLAINTEXT** for a normal cluster.

- **kafka.kerberos.domain.name**

  You do not need to set this parameter for a normal cluster. For a security cluster, the value of this parameter is the value of **kerberos.domain.name** in the Kafka cluster.

  For details, check **${BIGDATA_HOME}/FusionInsight_Current/1_X_Broker/etc/server.properties** on the node where the broker instance resides.

  In the preceding paths, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 10 The Flume client automatically loads the information in the **properties.properties** file.

After new log files are generated in the directory specified by **spoolDir**, the logs will be sent to Kafka producers and can be consumed by Kafka consumers.

----End

6.3.4 Using an HBase Client

Scenario

This section describes how to use the HBase client in an O&M scenario or a service scenario.

Prerequisites

- The client has been installed. For example, the installation directory is **/opt/hadoopclient**. The client directory in the following operations is only an example. Change it to the actual installation directory.
- Service component users are created by the administrator as required.

  A machine-machine user needs to download the **keytab** file and a human-machine user needs to change the password upon the first login.

- If a non-**root** user uses the HBase client, ensure that the owner of the HBase client directory is this user. Otherwise, run the following command to change the owner.

  chown user:group -R *Client installation directory*/**HBase**

Using the HBase Client (Versions Earlier Than MRS 3.x)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Component service user*

For example, **kinit hbaseuser**.

Step 5 Run the following HBase client command:

hbase shell

----End

Using the HBase Client (MRS 3.x or Later)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If you use the client to connect to a specific HBase instance in a scenario where multiple HBase instances are installed, run the following command to load the environment variables of the instance. Otherwise, skip this step. For example, to load the environment variables of the HBase2 instance, run the following command:

source HBase2/component_env

Step 5 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Component service user*

For example, **kinit hbaseuser**.

Step 6 Run the following HBase client command:

hbase shell

----End

Common HBase client commands

The following table lists common HBase client commands. For more commands, see **http://hbase.apache.org/2.2/book.html**.

Table 6-4 HBase client commands

Command	Description
create	Used to create a table, for example, **create 'test', 'f1', 'f2', 'f3'**.
disable	Used to disable a specified table, for example, **disable 'test'**.
enable	Used to enable a specified table, for example, **enable 'test'**.
alter	Used to alter the table structure. You can run the **alter** command to add, modify, or delete column family information and table-related parameter values, for example, **alter 'test', {NAME => 'f3', METHOD => 'delete'}**.
describe	Used to obtain the table description, for example, **describe 'test'**.
drop	Used to delete a specified table, for example, **drop 'test'**. Before deleting a table, you must stop it.
put	Used to write the value of a specified cell, for example, **put 'test','r1','f1:c1','myvalue1'**. The cell location is unique and determined by the table, row, and column.
get	Used to get the value of a row or the value of a specified cell in a row, for example, **get 'test','r1'**.
scan	Used to query table data, for example, **scan 'test'**. The table name and scanner must be specified in the command.

6.3.5 Using an HDFS Client

Scenario

This section describes how to use the HDFS client in an O&M scenario or service scenario.

Prerequisites

- The client has been installed.

  For example, the installation directory is **/opt/hadoopclient**. The client directory in the following operations is only an example. Change it to the actual installation directory.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user needs to change the password upon the first login. (This operation is not required in normal mode.)

Using the HDFS Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the HDFS Shell command. Example:

hdfs dfs -ls /

----End

Common HDFS Client Commands

The following table lists common HDFS client commands.

For more commands, see **https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/CommandsManual.html#User_Commands**.

Table 6-5 Common HDFS client commands

Command	Description	Example
hdfs dfs -mkdir *Folder name*	Used to create a folder.	**hdfs dfs -mkdir /tmp/mydir**
hdfs dfs -ls *Folder name*	Used to view a folder.	**hdfs dfs -ls /tmp**
hdfs dfs -put *Local file on the client node Specified HDFS path*	Used to upload a local file to a specified HDFS path.	**hdfs dfs -put /opt/test.txt /tmp** Upload the **/opt/test.txt** file on the client node to the **/tmp** directory of HDFS.
hdfs dfs -get *Specified file on HDFS Specified path on the client node*	Used to download the HDFS file to the specified local path.	**hdfs dfs -get /tmp/test.txt /opt/** Download the **/tmp/test.txt** file on HDFS to the **/opt** path on the client node.
hdfs dfs -rm -r -f *Specified folder on HDFS*	Used to delete a folder.	**hdfs dfs -rm -r -f /tmp/mydir**

Command	Description	Example
hdfs dfs -chmod *Permission parameter File directory*	Used to configure the HDFS directory permission for a user.	**hdfs dfs -chmod 700 /tmp/test**

Client-related FAQs

1. What do I do when the HDFS client exits abnormally and error message "java.lang.OutOfMemoryError" is displayed after the HDFS client command is running?

   This problem occurs because the memory required for running the HDFS client exceeds the preset upper limit (128 MB by default). You can change the memory upper limit of the client by modifying **CLIENT_GC_OPTS** in *<Client installation path>***/HDFS/component_env**. For example, if you want to set the upper limit to 1 GB, run the following command:

   ```
   CLIENT_GC_OPTS="-Xmx1G"
   ```

   After the modification, run the following command to make the modification take effect:

   source <*Client installation path*>/**/bigdata_env**

2. How do I set the log level when the HDFS client is running?

   By default, the logs generated during the running of the HDFS client are printed to the console. The default log level is INFO. To enable the DEBUG log level for fault locating, run the following command to export an environment variable:

   export HADOOP_ROOT_LOGGER=DEBUG,console

   Then run the HDFS Shell command to generate the DEBUG logs.

   If you want to print INFO logs again, run the following command:

   export HADOOP_ROOT_LOGGER=INFO,console

3. How do I delete HDFS files permanently?

   HDFS provides a recycle bin mechanism. Typically, after an HDFS file is deleted, the file is moved to the recycle bin of HDFS. If the file is no longer needed and the storage space needs to be released, clear the corresponding recycle bin directory, for example, **hdfs://hacluster/user/xxx/.Trash/Current/***xxx*.

6.3.6 Using a Hive Client

Scenario

This section guides users to use a Hive client in an O&M or service scenario.

Prerequisites

- The client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login.

Using the Hive Client (Versions Earlier Than MRS 3.x)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Log in to the Hive client based on the cluster authentication mode.

- In security mode, run the following command to complete user authentication and log in to the Hive client:

  kinit *Component service user*

  beeline

- In common mode, run the following command to log in to the Hive client. If no component service user is specified, the current OS user is used to log in to the Hive client.

  beeline -n *component service user*

  NOTE

  After a beeline connection is established, you can compile and submit HQL statements to execute related tasks. To run the Catalog client command, you need to run the **!q** command first to exit the beeline environment.

Step 5 Run the following command to execute the HCatalog client command:

hcat -e *"cmd"*

cmd must be a Hive DDL statement, for example, **hcat -e "show tables"**.

NOTE

- To use the HCatalog client, choose **More** > **Download Client** on the service page to download the clients of all services. This restriction does not apply to the beeline client.
- Due to permission model incompatibility, tables created using the HCatalog client cannot be accessed on the HiveServer client. However, the tables can be accessed on the WebHCat client.
- If you use the HCatalog client in Normal mode, the system performs DDL commands using the current user who has logged in to the operating system.
- Exit the beeline client by running the **!q** command instead of by pressing **Ctrl + c**. Otherwise, the temporary files generated by the connection cannot be deleted and a large number of junk files will be generated as a result.
- If multiple statements need to be entered during the use of beeline clients, separate the statements from each other using semicolons (**;**) and set the value of **entireLineAsCommand** to **false**.

  Setting method: If beeline has not been started, run the **beeline --entireLineAsCommand=false** command. If the beeline has been started, run the **!set entireLineAsCommand false** command.

  After the setting, if a statement contains semicolons (**;**) that do not indicate the end of the statement, escape characters must be added, for example, **select concat_ws('\;', collect_set(col1)) from tbl**.

----End

Using the Hive Client (MRS 3.x or Later)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 MRS 3.*X* supports multiple Hive instances. If you use the client to connect to a specific Hive instance in a scenario when multiple Hive instances are installed, run the following command to load the environment variables of the instance. Otherwise, skip this step. For example, load the environment variables of the Hive2 instance.

source Hive2/component_env

Step 5 Log in to the Hive client based on the cluster authentication mode.

- In security mode, run the following command to complete user authentication and log in to the Hive client:

  kinit *Component service user*

  beeline

- In common mode, run the following command to log in to the Hive client. If no component service user is specified, the current OS user is used to log in to the Hive client.

  beeline -n *component service user*

Step 6 Run the following command to execute the HCatalog client command:

hcat -e *"cmd"*

cmd must be a Hive DDL statement, for example, **hcat -e "show tables"**.

NOTE

- To use the HCatalog client, choose **More** > **Download Client** on the service page to download the clients of all services. This restriction does not apply to the beeline client.
- Due to permission model incompatibility, tables created using the HCatalog client cannot be accessed on the HiveServer client. However, the tables can be accessed on the WebHCat client.
- If you use the HCatalog client in Normal mode, the system performs DDL commands using the current user who has logged in to the operating system.
- Exit the beeline client by running the **!q** command instead of by pressing **Ctrl + C**. Otherwise, the temporary files generated by the connection cannot be deleted and a large number of junk files will be generated as a result.
- If multiple statements need to be entered during the use of beeline clients, separate the statements from each other using semicolons (**;**) and set the value of **entireLineAsCommand** to **false**.

  Setting method: If beeline has not been started, run the **beeline --entireLineAsCommand=false** command. If the beeline has been started, run the **!set entireLineAsCommand false** command.

  After the setting, if a statement contains semicolons (**;**) that do not indicate the end of the statement, escape characters must be added, for example, **select concat_ws('\;', collect_set(col1)) from tbl**.

----End

Common Hive Client Commands

The following table lists common Hive Beeline commands.

For more commands, see **https://cwiki.apache.org/confluence/display/Hive/HiveServer2+Clients#HiveServer2Clients-BeelineCommands**.

Table 6-6 Common Hive Beeline commands

Command	Description
set <key>=<value>	Sets the value of a specific configuration variable (key). **NOTE** If the variable name is incorrectly spelled, the Beeline does not display an error.
set	Prints the list of configuration variables overwritten by users or Hive.
set -v	Prints all configuration variables of Hadoop and Hive.
add FILE[S] <filepath> <filepath>*add JAR[S] <filepath> <filepath>*add ARCHIVE[S] <filepath> <filepath>*	Adds one or more files, JAR files, or ARCHIVE files to the resource list of the distributed cache.

Command	Description
add FILE[S] <ivyurl> <ivyurl>* add JAR[S] <ivyurl> <ivyurl>* add ARCHIVE[S] <ivyurl> <ivyurl>*	Adds one or more files, JAR files, or ARCHIVE files to the resource list of the distributed cache using the lvy URL in the **ivy://goup:module:version?query_string** format.
list FILE[S]list JAR[S]list ARCHIVE[S]	Lists the resources that have been added to the distributed cache.
list FILE[S] <filepath>*list JAR[S] <filepath>*list ARCHIVE[S] <filepath>*	Checks whether given resources have been added to the distributed cache.
delete FILE[S] <filepath>*delete JAR[S] <filepath>*delete ARCHIVE[S] <filepath>*	Deletes resources from the distributed cache.
delete FILE[S] <ivyurl> <ivyurl>* delete JAR[S] <ivyurl> <ivyurl>* delete ARCHIVE[S] <ivyurl> <ivyurl>*	Delete the resource added using **<ivyurl>** from the distributed cache.
reload	Enable HiveServer2 to discover the change of the JAR file **hive.reloadable.aux.jars.path** in the specified path. (You do not need to restart HiveServer2.) Change actions include adding, deleting, or updating JAR files.
dfs <dfs command>	Runs the **dfs** command.
<query string>	Executes the Hive query and prints the result to the standard output.

6.3.7 Using an Impala Client

Impala is a massively parallel processing (MPP) SQL query engine for processing vast amounts of data stored in Hadoop clusters. It is an open source software written in C++ and Java. It provides high performance and low latency compared with other SQL engines for Hadoop.

Background

Suppose a user develops an application to manage users who use service A in an enterprise. The procedure of operating service A on the Impala client is as follows:

Operations on common tables:

- Create the **user_info** table.
- Add users' educational backgrounds and titles to the table.
- Query user names and addresses by user ID.
- Delete the user information table after service A ends.

Table 6-7 User information

No.	Name	Gender	Age	Address
12005000201	A	Male	19	City A
12005000202	B	Female	23	City B
12005000203	C	Male	26	City C
12005000204	D	Male	18	City D
12005000205	E	Female	21	City E
12005000206	F	Male	32	City F
12005000207	G	Female	29	City G
12005000208	H	Female	30	City H
12005000209	I	Male	26	City I
12005000210	J	Female	25	City J

Prerequisites

The client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the Impala client command to implement service A.

Run the client command of the Impala component directly.

impala-shell

NOTE

By default, **impala-shell** attempts to connect to the Impala daemon on port 21000 of **localhost**. To connect to another host, use the **-i < host:port >** option, for example, **impala-shell -i** *xxx.xxx.xxx.xxx***:21000**. To automatically connect to a specific Impala database, use the **-d <database>** option. For example, if all your Kudu tables are in the **impala_kudu** database, **-d impala_kudu** can use this database. To exit the Impala Shell, run the **quit** command.

Operations on internal tables:

1. Create the **user_info** user information table according to **Table 6-7** and add data to it.
   ```
   create table user_info(id string,name string,gender string,age int,addr string);
   insert into table user_info(id,name,gender,age,addr) values("12005000201", "A", "Male", 19, "City A");
   ```
   ... (Other statements are the same.)
2. Add users' educational backgrounds and titles to the **user_info** table.

   For example, to add educational background and title information about user 12005000201, run the following commands.
   ```
   alter table user_info add columns(education string,technical string);
   ```
3. Query user names and addresses by user ID.

   For example, to query the name and address of user 12005000201, run the following command:
   ```
   select name,addr from user_info where id='12005000201';
   ```
4. Delete the user information table:
   ```
   drop table user_info;
   ```

Operations on external partition tables:

Create an external partition table and import data.

1. Create a path for storing external table data.
   - Security mode (Kerberos authentication is enabled for clusters)

     cd /opt/hadoopclient

     source bigdata_env

     kinit *hive*

     NOTE

     The user must have the hive administrator permissions.

     impala-shell

     hdfs dfs -mkdir /hive

     hdfs dfs -mkdir /hive/user_info
   - Normal mode (Kerberos authentication is disabled for clusters)

     su - omm

     cd /opt/hadoopclient

     source bigdata_env

     impala-shell

     hdfs dfs -mkdir /hive

     hdfs dfs -mkdir /hive/user_info

2. Create a table.
   ```
   create external table user_info(id string,name string,gender string,age int,addr string) partitioned by(year string) row format delimited fields terminated by ' ' lines terminated by '\n' stored as textfile location '/hive/user_info';
   ```

   NOTE

   fields terminated indicates delimiters, for example, spaces.

   lines terminated indicates line breaks, for example, **\n**.

   /hive/user_info indicates the path of the data file.

3. Import data.
   a. Execute the **insert** statement to insert data.
      ```
      insert into user_info partition(year="2018") values ("12005000201", "A", "Male", 19, "City A");
      ```
   b. Run the **load data** command to import file data.
      i. Create a file based on the data in **Table 6-7**. For example, the file name is **txt.log**. Fields are separated by space, and the line feed characters are used as the line breaks.
      ii. Upload the file to HDFS.

      hdfs dfs -put txt.log /tmp

      iii. Load data to the table.

      load data inpath '/tmp/txt.log' into table user_info partition (year='2018');

4. Query the imported data:
   ```
   select * from user_info;
   ```
5. Delete the user information table:
   ```
   drop table user_info;
   ```

----End

6.3.8 Using a Kafka Client

Scenario

You can create, query, and delete topics on a cluster client.

Prerequisites

The client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Using the Kafka Client (Versions Earlier Than MRS 3.x)

Step 1 Access the ZooKeeper instance page.

Click the cluster name to go to the cluster details page and choose **Components** > **ZooKeeper** > **Instances**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 2 View the IP addresses of the ZooKeeper role instance.

Record any IP address of the ZooKeeper instance.

Step 3 Log in to the node where the client is installed.

Step 4 Run the following command to switch to the client directory, for example, **/opt/hadoopclient/Kafka/kafka/bin**.

cd /opt/hadoopclient/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/hadoopclient/bigdata_env

Step 6 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Kafka user*

Step 7 Create a topic.

sh kafka-topics.sh --create --topic *Topic name* **--partitions** *Number of partitions occupied by the topic* **--replication-factor** *Number of replicas of the topic* **--zookeeper** *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

Step 8 Run the following command to view the topic information in the cluster:

sh kafka-topics.sh --list --zookeeper *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

Step 9 Delete the topic created in **Step 7**.

sh kafka-topics.sh --delete --topic *Topic name* **--zookeeper** *IP address of the node where the ZooKeeper instance resides*:*clientPort*/**kafka**

Type **y** and press **Enter**.

----End

Using the Kafka Client (MRS 3.x or Later)

Step 1 Access the ZooKeeper instance page.

Log in to FusionInsight Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Instance**.

Step 2 View the IP addresses of the ZooKeeper role instance.

Record any IP address of the ZooKeeper instance.

Step 3 Log in to the node where the client is installed.

Step 4 Run the following command to switch to the client directory, for example, **/opt/hadoopclient/Kafka/kafka/bin**.

cd /opt/hadoopclient/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/hadoopclient/bigdata_env

Step 6 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Kafka user*

Step 7 Log in to FusionInsight Manager, choose **Cluster** > **Name of the desired cluster** > **Services** > **ZooKeeper**, and click the **Configurations** tab and then **All Configurations**. On the displayed page, search for the **clientPort** parameter and record its value.

Step 8 Create a topic.

sh kafka-topics.sh --create --topic *Topic name* **--partitions** *Number of partitions occupied by the topic* **--replication-factor** *Number of replicas of the topic* **--zookeeper** *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

Step 9 Run the following command to view the topic information in the cluster:

sh kafka-topics.sh --list --zookeeper *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

Step 10 Delete the topic created in **Step 8**.

sh kafka-topics.sh --delete --topic *Topic name* **--zookeeper** *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

----End

6.3.9 Using a Kudu Client

Kudu is a columnar storage manager developed for the Apache Hadoop platform. Kudu shares the common technical properties of Hadoop ecosystem applications. It is horizontally scalable and supports highly available operations.

Prerequisites

The cluster client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the Kudu command line tool.

Run the command line tool of the Kudu component to view help information.

kudu -h

The command output is as follows:

```
Usage: kudu <command> [<args>]

<command> can be one of the following:
        cluster   Operate on a Kudu cluster
       diagnose   Diagnostic tools for Kudu servers and clusters
             fs   Operate on a local Kudu filesystem
            hms   Operate on remote Hive Metastores
  local_replica   Operate on local tablet replicas via the local filesystem
         master   Operate on a Kudu Master
            pbc   Operate on PBC (protobuf container) files
           perf   Measure the performance of a Kudu cluster
 remote_replica   Operate on remote tablet replicas on a Kudu Tablet Server
          table   Operate on Kudu tables
         tablet   Operate on remote Kudu tablets
           test   Various test actions
        tserver   Operate on a Kudu Tablet Server
            wal   Operate on WAL (write-ahead log) files
```

NOTE

The Kudu command line tool does not support DDL and DML operations, but provides the refined query function for the **cluster**, **master**, **tserver**, **fs**, and **table** parameters.

Common operations:

- Check the tables in the current cluster.

  kudu table list *KuduMaster instance IP1:7051, KuduMaster instance IP2:7051, KuduMaster instance IP3:7051*

- Query the configurations of the KuduMaster instance of the Kudu service.

  kudu master get_flags *KuduMaster instance IP:7051*

- Query the schema of a table.

  kudu table describe *KuduMaster instance IP1:7051, KuduMaster instance IP2:7051, KuduMaster instance IP3:7051 Table name*

- Delete a table.

  kudu table delete *KuduMaster instance IP1:7051, KuduMaster instance IP2:7051, KuduMaster instance IP3:7051 Table name*

  NOTE

  To obtain the IP address of the KuduMaster instance, choose **Components** > **Kudu** > **Instances** on the cluster details page.

----End

6.3.10 Using the Oozie Client

Scenario

This section describes how to use the Oozie client in an O&M scenario or service scenario.

Prerequisites

- The client has been installed. For example, the installation directory is **/opt/client**. The client directory in the following operations is only an example.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login.

Using the Oozie Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to switch to the client installation directory (change it to the actual installation directory):

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Check the cluster authentication mode.

- If the cluster is in security mode, run the following command to authenticate the user: *exampleUser* indicates the name of the user who submits tasks.

  kinit *exampleUser*
- If the cluster is in normal mode, go to **Step 5**.

Step 5 Perform the following operations to configure Hue:

1. Configure the Spark2x environment (skip this step if the Spark2x task is not involved):

   hdfs dfs -put /opt/client/Spark2x/spark/jars/*.jar /user/oozie/share/lib/spark2x/

   When the JAR package in the HDFS directory **/user/oozie/share** changes, you need to restart the Oozie service.
2. Upload the Oozie configuration file and JAR package to HDFS.

   hdfs dfs -mkdir /user/*exampleUser*

   hdfs dfs -put -f /opt/client/Oozie/oozie-client-*/examples /user/*exampleUser*/

NOTE

- *exampleUser* indicates the name of the user who submits tasks.
- If the user who submits the task and other files except **job.properties** are not changed, client installation directory **Oozie/oozie-client-*/examples** can be repeatedly used after being uploaded to HDFS.
- Resolve the JAR file conflict between Spark and Yarn about Jetty.

  hdfs dfs -rm -f /user/oozie/share/lib/spark/jetty-all-9.2.22.v20170606.jar
- In normal mode, if **Permission denied** is displayed during the upload, run the following commands:

  su - omm

  source /opt/client/bigdata_env

  hdfs dfs -chmod -R 777 /user/oozie

  exit

----End

6.3.11 Using a Storm Client

Scenario

This section describes how to use the Storm client in an O&M scenario or service scenario.

Prerequisites

- You have installed the client. For example, the installation directory is **/opt/hadoopclient**.
- Service component users are created by the administrator as required. In security mode, machine-machine users have downloaded the keytab file. A human-machine user must change the password upon the first login. (Not involved in normal mode)

Procedure

Step 1 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see **Using an MRS Client**.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If multiple Storm instances are installed, run the following command to load the environment variables of a specific instance when running the Storm command to submit the topology. Otherwise, skip this step. The following command uses the instance Storm-2 as an example.

source Storm-2/component_env

Step 5 Run the following command to perform user authentication (skip this step in normal mode):

kinit *Component service user*

Step 6 Run the following command to perform operations on the client:

For example, run the following command:

- **cql**
- **storm**

NOTE

A Storm client cannot be connected to secure and non-secure ZooKeepers at the same time.

----End

6.3.12 Using a Yarn Client

Scenario

This section guides users to use a Yarn client in an O&M or service scenario.

Prerequisites

- The client has been installed.

  For example, the installation directory is **/opt/hadoopclient**. The client directory in the following operations is only an example. Change it to the actual installation directory.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login. In common mode, you do not need to download the keytab file or change the password.

Using the Yarn Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the Yarn command. The following provides an example:

yarn application -list

----End

Client-related FAQs

1. What Do I Do When the Yarn Client Exits Abnormally and Error Message "java.lang.OutOfMemoryError" Is Displayed After the Yarn Client Command Is Run?

   This problem occurs because the memory required for running the Yarn client exceeds the upper limit (128 MB by default) set on the Yarn client. For clusters of MRS 3.x or later: You can modify **CLIENT_GC_OPTS** in *<Client installation path>***/HDFS/component_env** to change the memory upper limit of the Yarn client. For example, if you want to set the maximum memory to 1 GB, run the following command:

   ```
   export CLIENT_GC_OPTS="-Xmx1G"
   ```

   For clusters earlier than MRS 3.x: You can modify **GC_OPTS_YARN** in *< Client installation path >***/HDFS/component_env** to change the memory upper limit of the Yarn client. For example, if you want to set the maximum memory to 1 GB, run the following command:

   ```
   export GC_OPTS_YARN="-Xmx1G"
   ```

   After the modification, run the following command to make the modification take effect:

   source <*Client installation path*>/**/bigdata_env**

2. How Can I Set the Log Level When the Yarn Client Is Running?

   By default, the logs generated during the running of the Yarn client are printed to the console. The default log level is INFO. To enable the DEBUG log level for fault locating, run the following command to export an environment variable:

   export YARN_ROOT_LOGGER=DEBUG,console

   Then run the Yarn Shell command to print DEBUG logs.

   If you want to print INFO logs again, run the following command:

   export YARN_ROOT_LOGGER=INFO,console

7 Configuring a Cluster with Storage and Compute Decoupled

7.1 Introduction to Storage-Compute Decoupling

In scenarios that require large storage capacity and elastic compute resources, MRS enables you to store data in OBS and use an MRS cluster for data computing only. In this way, storage and compute are separated.

NOTE

In the big data decoupled storage-compute scenario, the OBS parallel file system must be used to configure a cluster. Using common object buckets will greatly affect the cluster performance.

Process of using the storage-compute decoupling function:

1. Configure a storage-compute decoupled cluster using either of the following methods (agency is recommended):
   - Bind an agency of the ECS type to an MRS cluster to access OBS, preventing the AK/SK from being exposed in the configuration file. For details, see **Configuring a Storage-Compute Decoupled Cluster (Agency)**.
   - Configure the AK/SK in an MRS cluster. The AK/SK will be exposed in the configuration file in plaintext. Exercise caution when performing this operation. For details, see **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.
2. Use the cluster.

   For details, see the following sections':
   - **Interconnecting Flink with OBS**
   - **Interconnecting Flume with OBS**
   - **Interconnecting HDFS with OBS**
   - **Interconnecting Hive with OBS**
   - **Interconnecting MapReduce with OBS**

  - **Interconnecting Spark2x with OBS**
  - **Interconnecting Sqoop with External Storage Systems**

7.2 Configuring a Storage-Compute Decoupled Cluster (Agency)

MRS allows you to store data in OBS and use an MRS cluster for data computing only. In this way, storage and compute are separated. You can create an IAM agency, which enables ECS to automatically obtain the temporary AK/SK to access OBS. This prevents the AK/SK from being exposed in the configuration file.

By binding an agency, ECSs or BMSs can manage some of your resources. Determine whether to configure an agency based on the actual service scenario.

MRS provides the following configuration modes for accessing OBS. You can select one of them. The agency mode is recommended.

- Bind an agency of the ECS type to an MRS cluster to access OBS, preventing the AK/SK from being exposed in the configuration file. For details, see the following part in this section.
- Configure the AK/SK in an MRS cluster. The AK/SK will be exposed in the configuration file in plaintext. Exercise caution when performing this operation. For details, see **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

This function is available for components Hadoop, Hive, Spark, Presto, and Flink in clusters of .

(Optional) Step 1: Create an ECS Agency with OBS Access Permissions

NOTE

- MRS presets **MRS_ECS_DEFAULT_AGENCY** in the agency list of IAM so that you can select this agency when creating a cluster. This agency has the **OBSOperateAccess** permission and the **CESFullAccess** (only available for users who have enabled fine-grained policies), **CES Administrator**, and **KMS Administrator** permissions in the region where the cluster is located. Do not modify **MRS_ECS_DEFAULT_AGENCY** on IAM.
- If you want to use the preset agency, skip the step for creating an agency. If you want to use a custom agency, perform the following steps to create an agency. (To create or modify an agency, you must have the Security Administrator permission.)

1. Log in to the management console.
2. Choose **Service List** > **Management & Governance** > **Identity and Access Management**.
3. Choose **Agencies**. On the displayed page, click **Create Agency**.
4. Enter an agency name, for example, **mrs_ecs_obs**.
5. Set **Agency Type** to **Cloud service** and select **ECS BMS** to authorize ECS or BMS to invoke OBS.
6. Set **Validity Period** to **Unlimited** and click **Next**.
7. On the displayed page, search for the **OBS OperateAccess** and select it.

8. Click **Next**. On the displayed page, select the desired scope for permissions you selected. By default, **All resources** is selected. Click **Show More** and select **Global resources**.
9. In the dialog box that is displayed, click **OK** to start authorization. After the message "**Authorization successful.**" is displayed, click **Finish**. The agency is successfully created.

Step 2: Create a Cluster with Storage and Compute Separated

You can configure an agency when creating a cluster or bind an agency to an existing cluster to separate storage and compute. This section uses a cluster with Kerberos authentication enabled as an example.

Configuring an agency when creating a cluster:

1. Log in to the MRS management console.
2. Click **Create Cluster**. The page for creating a cluster is displayed.
3. Click the **Custom Config** tab.
4. On the **Custom Config** tab page, set software parameters.
   - **Region**: Select a region as required.
   - **Cluster Name**: You can use the default name. However, you are advised to include a project name abbreviation or date for consolidated memory and easy distinguishing.
   - Cluster Version: Select a cluster version.
   - **Cluster Type**: Select **Analysis cluster** or **Hybrid cluster** and select all components.
   - **Metadata**: Select **Local**.
5. Click **Next** and set hardware parameters.
   - **AZ**: Use the default value.
   - **VPC**: Use the default value.
   - **Subnet**: Use the default value.
   - **Security Group**: Use the default value.
   - **EIP**: Use the default value.
   - **Enterprise Project**: Use the default value.
   - **Cluster Node**: Select the number of cluster nodes and node specifications based on site requirements.
6. Click **Next** and set related parameters.
   - **Kerberos Authentication**: This function is enabled by default. You can enable or disable it.
   - **Username**: The default username is **admin**, which is used to log in to MRS Manager.
   - **Password**: Set a password for user **admin**.
   - **Confirm Password**: Enter the password of user **admin** again.
   - **Login Mode**: Select a method for logging in to ECSs. In this example, select **Password**.
   - **Username**: The default username is **root**, which is used to remotely log in to ECSs.

- **Password**: Set a password for user **root**.
- **Confirm Password**: Enter the password of user **root** again.

7. In this example, configure an agency and leave other parameters blank. For details about how to configure other parameters, see **(Optional) Advanced Configuration**.

   Agency: Select the agency created in **(Optional) Step 1: Create an ECS Agency with OBS Access Permissions** or **MRS_ECS_DEFAULT_AGENCY** preset in IAM.

8. To enable secure communications, select **Enable**. For details, see **Communication Security Authorization**.
9. Click and wait until the cluster is created.

   If Kerberos authentication is enabled for a cluster, check whether Kerberos authentication is required. If yes, click **Continue**. If no, click **Back** to disable Kerberos authentication and then create a cluster.

Configuring an agency for an existing cluster:

1. Log in to the MRS management console. In the left navigation pane, choose **Clusters** > **Active Clusters**.
2. Click the name of the cluster to enter its details page.
3. On the **Dashboard** page, click **Synchronize** on the right of **IAM User Sync** to synchronize IAM users.
4. On the **Dashboard** tab page, click **Manage Agency** on the right side of **Agency** to select an agency and click **OK** to bind it. Alternatively, click **Create Agency** to go to the IAM console to create an agency and select it.

Step 3: Create an OBS File System for Storing Data

NOTE

In the big data decoupled storage-compute scenario, the OBS parallel file system must be used to configure a cluster. Using common object buckets will greatly affect the cluster performance.

1. Log in to OBS Console.
2. Choose **Parallel File System** > **Create Parallel File System**.
3. Enter the file system name, for example, **mrs-word001**.

   Set other parameters as required.

4. Click **Create Now**.
5. In the parallel file system list on the OBS console, click the file system name to go to the details page.
6. In the navigation pane, choose **Files** and create the **program** and **input** folders.
   - **program**: Upload the program package to this folder.
   - **input**: Upload the input data to this folder.

Step 4: Accessing the OBS File System

1. Log in to a Master node as user **root**. For details, see **Logging In to an ECS**.

2. Run the following command to set the environment variables:

   For versions earlier than MRS 3.x, run the **source /opt/client/bigdata_env** command.

   For MRS 3.x or later, run the **source /opt/Bigdata/client/bigdata_env** command.

3. Verify that Hadoop can access OBS.

   a. View the list of files in the file system **mrs-word001**.

      hadoop fs -ls obs://mrs-word001/

   b. Check whether the file list is returned. If it is returned, OBS access is successful.

      Figure 7-1 Returned file list

      ```
      Found 2 items
      drwxrwxrwx   - root root          0 2019-12-21 11:04 obs://mrs-word001/input
      drwxrwxrwx   - root root          0 2019-12-21 11:04 obs://mrs-word001/program
      ```

4. Verify that Hive can access OBS.

   a. If Kerberos authentication has been enabled for the cluster, run the following command to authenticate the current user. The current user must have a permission to create Hive tables. For details about how to configure a role with a permission to create Hive tables, see **Creating a Role**. For details about how to create a user and bind a role to the user, see **Creating a User**. If Kerberos authentication is disabled for the current cluster, skip this step.

      kinit *MRS cluster user*

      Example: **kinit hiveuser**

   b. Run the client command of the Hive component.

      beeline

   c. Access the OBS directory in the beeline. For example, run the following command to create a Hive table and specify that data is stored in the **test_obs** directory of the file system **mrs-word001**:

      create table test_obs(a int, b string) row format delimited fields terminated by "," stored as textfile location "obs://mrs-word001/test_obs";

   d. Run the following command to query all tables. If table **test_obs** is displayed in the command output, OBS access is successful.

      show tables;

      Figure 7-2 Returned table name

      ```
      +-----------+
      | tab_name  |
      +-----------+
      | test_obs  |
      +-----------+
      1 row selected (0.352 seconds)
      ```

   e. Press **Ctrl+C** to exit the Hive beeline.

5. Verify that Spark can access OBS.

a. Run the client command of the Spark component.

spark-beeline

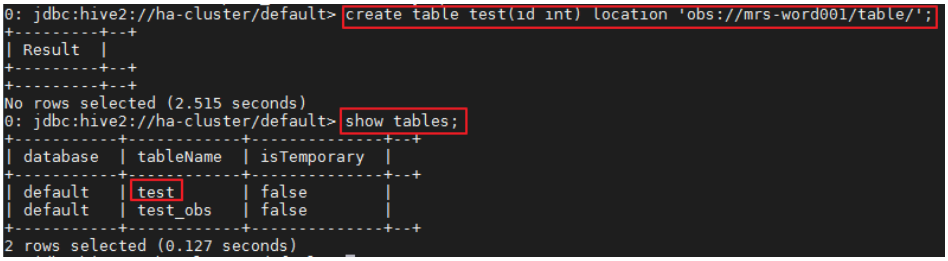
b. Access OBS in spark-beeline. For example, create table **test** in the **obs://mrs-word001/table/** directory.

create table test(id int) location 'obs://mrs-word001/table/';

c. Run the following command to query all tables. If table **test** is displayed in the command output, OBS access is successful.

show tables;

Figure 7-3 Returned table name



d. Press **Ctrl+C** to exit the Spark beeline.

6. Verify that Presto can access OBS.

– For normal clusters with Kerberos authentication disabled

i. Run the following command to connect to the client:

presto_cli.sh

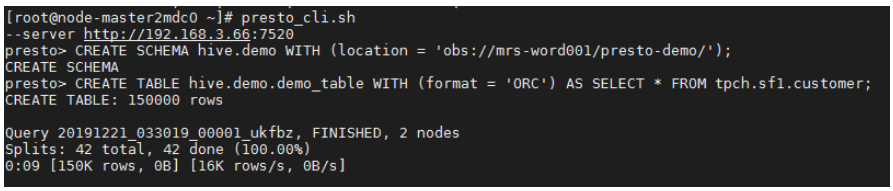
ii. On the Presto client, run the following statement to create a schema and set **location** to an OBS path:

CREATE SCHEMA hive.demo01 WITH (location = 'obs://mrs-word001/presto-demo002/');

iii. Create a table in the schema. The table data is stored in the OBS file system. The following is an example.

CREATE TABLE hive.demo.demo_table WITH (format = 'ORC') AS SELECT * FROM tpch.sf1.customer;

Figure 7-4 Return result



iv. Run **exit** to exit the client.

– For security clusters with Kerberos authentication enabled

i. Log in to MRS Manager and create a role with the Hive Admin Privilege permissions, for example, **prestorole**. For details about how to create a role, see **Creating a Role**.

ii. Create a user that belongs to the Presto and Hive groups and bind the role created in **6.i** to the user, for example, **presto001**. For details about how to create a user, see **Creating a User**.

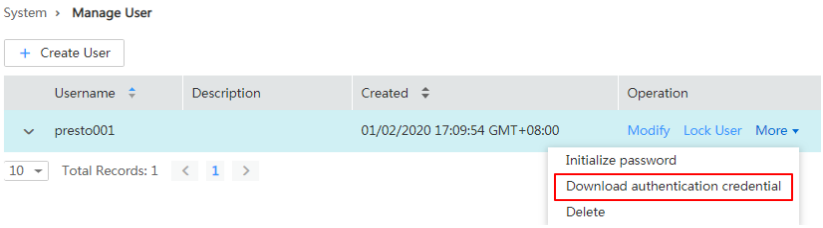
iii. Authenticate the current user.

kinit presto001

iv. Download the user credential.

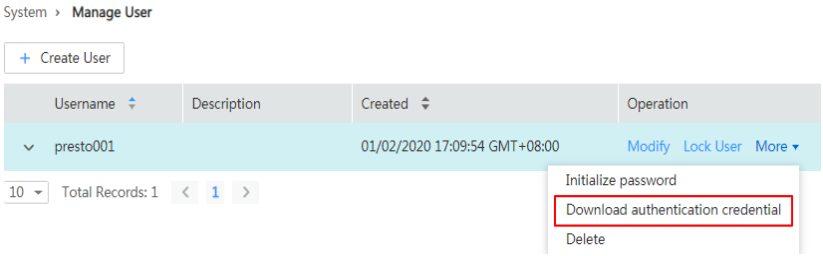
1) For MRS 3.x earlier, on MRS Manager, choose **System** > **Manage User**. In the row of the new user, choose **More** > **Download Authentication Credential**.

Figure 7-5 Downloading the Presto user authentication credential



2) On FusionInsight Manager for MRS 3.x or later,, choose **System > Permission > User**. In the row that contains the newly added user, click **More > Download Authentication Credential**.

Figure 7-6 Downloading the Presto user authentication credential



v. Decompress the downloaded user credential file, and save the obtained **krb5.conf** and **user.keytab** files to the client directory, for example, **/opt/Bigdata/client/Presto/**.

vi. Run the following command to obtain a user principal:

klist -kt /opt/Bigdata/client/Presto/user.keytab

vii. For clusters with Kerberos authentication enabled, run the following command to connect to the Presto Server of the cluster:

presto_cli.sh --krb5-config-path {krb5.conf file path} --krb5-principal {user principal} --krb5-keytab-path {user.keytab file path} --user {presto username}

- **krb5.conf** file path: Replace it with the file path set in **6.v**, for example, **/opt/Bigdata/client/Presto/krb5.conf**.
- **user.keytab** file path: Replace it with the file path set in **6.v**, for example, **/opt/Bigdata/client/Presto/user.keytab**.
- **user principal**: Replace it with the result returned in **6.vi**.
- **presto username**: Replace it with the name of the user created in **6.ii**, for example, **presto001**.

Example: presto_cli.sh --krb5-config-path /opt/Bigdata/client/Presto/krb5.conf --krb5-principal prest001@xxx_xxx_xxx_xxx.COM --krb5-keytab-path /opt/Bigdata/client/Presto/user.keytab --user presto001

viii. On the Presto client, run the following statement to create a schema and set **location** to an OBS path:

CREATE SCHEMA hive.demo01 WITH (location = 'obs://mrs-word001/presto-demo002/');

ix. Create a table in the schema. The table data is stored in the OBS file system. The following is an example.

CREATE TABLE hive.demo01.demo_table WITH (format = 'ORC') AS SELECT * FROM tpch.sf1.customer;

Figure 7-7 Return result

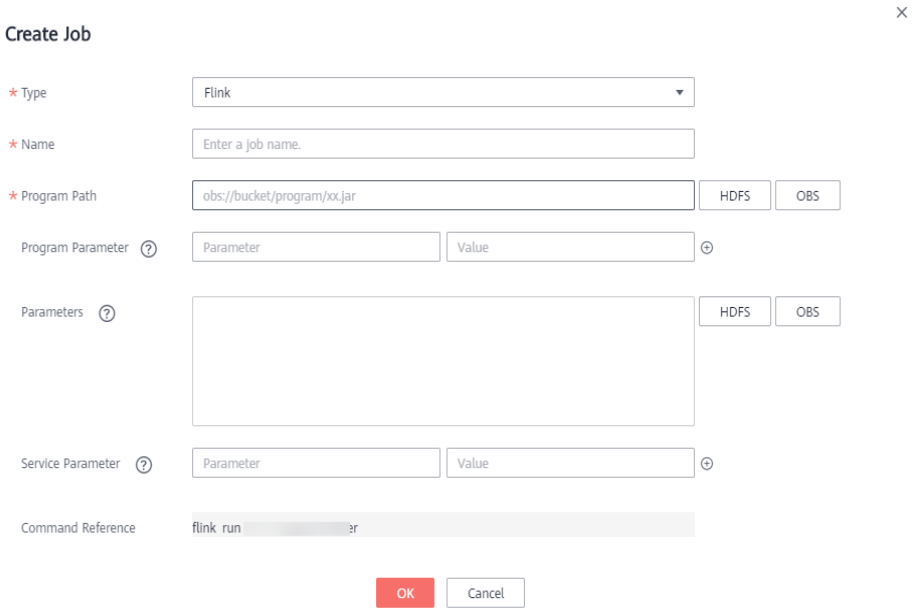
x. Run **exit** to exit the client.

7. Verify that Flink can access OBS.

a. On the **Dashboard** page, click **Synchronize** on the right of **IAM User Sync** to synchronize IAM users.

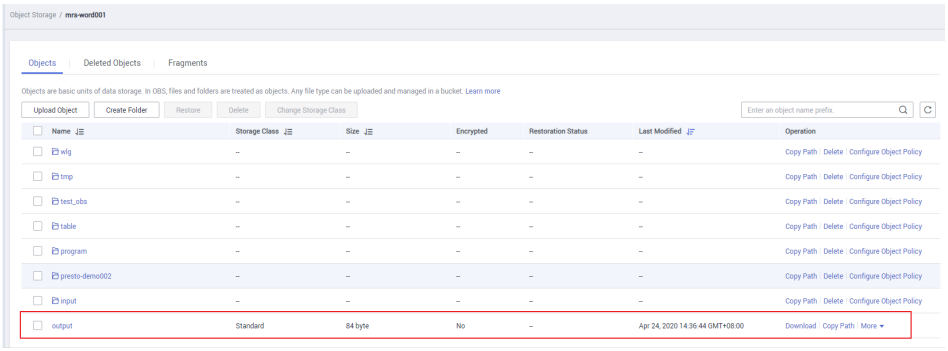
b. After user synchronization is complete, choose **Jobs** > **Create** on the cluster details page to create a Flink job. In **Parameters**, enter parameters in **--input <Job input path> --output <Job output path>** format. You can click **OBS** to select a job input path, and enter a job output path that does not exist, for example, **obs://mrs-word001/output/**. See **Figure 7-8**.

Figure 7-8 Creating a Flink job



c. On OBS Console, go to the output path specified during job creation. If the output directory is automatically created and contains the job execution results, OBS access is successful.

Figure 7-9 Flink job execution result



Reference

For details about how to control permissions to access OBS, see **Configuring Fine-Grained Permissions for MRS Multi-User Access to OBS**.

7.3 Configuring a Storage-Compute Decoupled Cluster (AK/SK)

In or later, OBS can be interconnected with MRS using **obs://**. Currently, Hadoop, Hive, Spark, Presto, and Flink are supported. HBase cannot use **obs://** to interconnect with OBS.

MRS provides the following configuration modes for accessing OBS. You can select one of them. The agency mode is recommended.

- Bind an agency of the ECS type to an MRS cluster to access OBS, preventing the AK/SK from being exposed in the configuration file. For details, see **Configuring a Storage-Compute Decoupled Cluster (Agency)**.
- Configure the AK/SK in an MRS cluster. The AK/SK will be exposed in the configuration file in plaintext. Exercise caution when performing this operation. For details, see the following part in this section.

NOTE

- To improve data write performance, change the value of the **fs.obs.buffer.dir** parameter of the corresponding service to a data disk directory.
- In the big data decoupled storage-compute scenario, the OBS parallel file system must be used to configure a cluster. Using common object buckets will greatly affect the cluster performance.

Using Hadoop to Access OBS

- Add the following content to file **core-site.xml** in the HDFS directory (**$client_home/ HDFS/hadoop/etc/hadoop**) on the MRS client:

```
<property>
  <name>fs.obs.access.key</name>
  <value>ak</value>
```

```
</property>
<property>
    <name>fs.obs.secret.key</name>
    <value>sk</value>
</property>
<property>
    <name>fs.obs.endpoint</name>
    <value>obs endpoint</value>
</property>
```

NOTICE

AK and SK will be displayed as plaintext in the configuration file. Exercise caution when setting AK and SK in the file.

After the configuration is added, you can directly access data on OBS without manually adding the AK/SK and endpoint. For example, run the following command to view the file list of the **test_obs_orc** directory in the **obs-test** file system:

hadoop fs -ls "obs://obs-test/test_obs_orc"

- Add AK/SK and endpoint to the command line to access data on OBS.

  hadoop fs -Dfs.obs.endpoint=xxx -Dfs.obs.access.key=xx -Dfs.obs.secret.key=xx -ls "obs://obs-test/ test_obs_orc"

Using Hive to Access OBS

Step 1 Log in to the service configuration page.

- For versions earlier than MRS 3.x, log in to the cluster details page and choose **Components** > **Hive** > **Service Configuration**.
- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > **Services** > **Hive** > **Configurations**.

Step 2 In the configuration type drop-down box, switch **Basic Configurations** to **All Configurations**.

Step 3 Search for **fs.obs.access.key** and **fs.obs.secret.key** and set them to the AK and SK of OBS respectively.

If the preceding two parameters cannot be found in the current cluster, choose **Hive** > **Customization** in the navigation tree on the left and add the two parameters to the customized parameter **core.site.customized.configs**.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. to restart the Hive service.

Step 5 Access the OBS directory in the beeline. For example, run the following command to create a Hive table and specify that data is stored in the **test_obs** directory in the **test-bucket** file system:

create table test_obs(a int, b string) row format delimited fields terminated by "," stored as textfile location "obs://test-bucket/test_obs";

----End

Using Spark to Access OBS

NOTE

SparkSQL depends on Hive. Therefore, when configuring OBS on Spark, you need to modify the OBS configuration used in **Using Hive to Access OBS**.

- spark-beeline and spark-sql

  You can add the following OBS attributes to the shell to access OBS:

  ```
  set fs.obs.endpoint=xxx
  set fs.obs.access.key=xxx
  set fs.obs.secret.key=xxx
  ```

- spark-beeline

  The spark-beeline can access OBS by configuring service parameters on Manager. The procedure is as follows:

  a. Log in to the service configuration page.

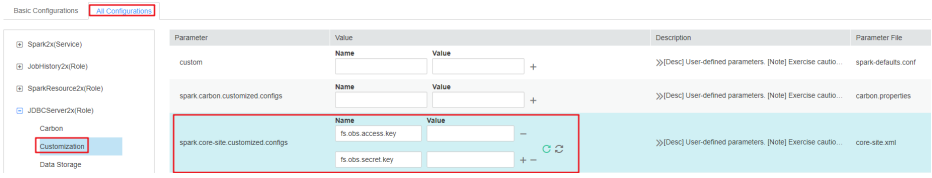
     - For versions earlier than MRS 3.x, log in to the cluster details page and choose **Components > Spark > Service Configuration**.
     - For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster > Services > Spark2x > Configurations**.

  b. In the configuration type drop-down box, switch **Basic Configurations** to **All Configurations**.

  c. Choose **JDBCServer** > **OBS**, and set values for **fs.obs.access.key** and **fs.obs.secret.key**.

     If the preceding two parameters cannot be found in the current cluster, choose **JDBCServer** > **Customization** in the navigation tree on the left and add the two parameters to the customized parameter **spark.core-site.customized.configs**.

     Figure 7-10 Parameters for adding an OBS

  d. Click **Save Configuration** and select **Restart the affected services or instances**. Restart the Spark service.

  e. Access OBS in **spark-beeline**. For example, access the **obs://obs-demo-input/table/** directory.

     create table test(id int) location 'obs://obs-demo-input/table/';

- spark-sql and spark-submit

  The spark-sql can also access OBS by modifying the **core-site.xml** configuration file.

  The method of modifying the configuration file is the same when you use the spark-sql and spark-submit to submit a task to access OBS.

  Add the following content to **core-site.xml** in the Spark configuration folder (**$client_home/Spark/spark/conf**) on the MRS client:

```
<property>
   <name>fs.obs.access.key</name>
   <value>ak</value>
</property>
<property>
   <name>fs.obs.secret.key</name>
   <value>sk</value>
</property>
<property>
   <name>fs.obs.endpoint</name>
   <value>obs endpoint</value>
</property>
```

Using Presto to Access OBS

Step 1 Go to the cluster details page and choose **Components** > **Presto** > **Service Configuration**.

Step 2 In the configuration type drop-down box, switch **Basic Configurations** to **All Configurations**.

Step 3 Search for and configure the following parameters:

- Set **fs.obs.access.key** to **AK**.
- Set **fs.obs.secret.key** to **SK**.

If the preceding two parameters cannot be found in the current cluster, choose **Presto > Hive** in the navigation tree on the left and add the two parameters to the customized parameter **core.site.customized.configs**.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**. to restart the Presto service.

Step 5 Choose **Components** > **Hive** > **Service Configuration**.

Step 6 In the configuration type drop-down box, switch **Basic Configurations** to **All Configurations**.

Step 7 Search for and configure the following parameters:

- Set **fs.obs.access.key** to **AK**.
- Set **fs.obs.secret.key** to **SK**.

Step 8 Click **Save Configuration** and select **Restart the affected services or instances**. to restart the Hive service.

Step 9 On the Presto client, run the following statement to create a schema and set **location** to an OBS path:

CREATE SCHEMA hive.demo WITH (location = 'obs://obs-demo/presto-demo/');

Step 10 Create a table in the schema. The table data is stored in the OBS file system. The following is an example.

CREATE TABLE hive.demo.demo_table WITH (format = 'ORC') AS SELECT * FROM tpch.sf1.customer;

----End

Using Flink to Access OBS

Add the following configuration to the Flink configuration file of the MRS client in ***Client installation path*/Flink/flink/conf/flink-conf.yaml**:

```
fs.obs.access.key:ak
fs.obs.secret.key: sk
fs.obs.endpoint: obs endpoint
```

NOTICE

AK and SK will be displayed as plaintext in the configuration file. Exercise caution when setting AK and SK in the file.

After the configuration is added, you can directly access data on OBS without manually adding the AK/SK and endpoint.

7.4 Using a Storage-Compute Decoupled Cluster

7.4.1 Interconnecting Flink with OBS

Before performing the following operations, ensure that you have configured a storage-compute decoupled cluster by referring to **Configuring a Storage-Compute Decoupled Cluster (Agency)** or **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

Step 1 Log in to the Flink client installation node as the client installation user.

Step 2 Run the following command to initialize environment variables:

source ${client_home}/bigdata_env

Step 3 Configure the Flink client properly. For details, see **Installing a Client (Version 3.x or Later)**.

Step 4 For a security cluster, run the following command to perform user authentication. If Kerberos authentication is not enabled for the current cluster, you do not need to run this command.

kinit *Username*

Step 5 Explicitly add the OBS file system to be accessed in the Flink command line.

./bin/flink run --class com.xxx.bigdata.flink.examples.FlinkProcessingTimeAPIMain ./config/FlinkCheckpointJavaExample.jar --chkPath obs://*Name of the OBS parallel file system*

----End

NOTE

Flink jobs are running on Yarn. Before configuring Flink to interconnect with the OBS file system, ensure that the interconnection between Yarn and the OBS file system is normal.

7.4.2 Interconnecting Flume with OBS

This section applies to MRS 3.x or later.

Before performing the following operations, ensure that you have configured a storage-compute decoupled cluster by referring to **Configuring a Storage-Compute Decoupled Cluster (Agency)** or **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

Step 1 Configure an agency.

1. Log in to the MRS console. In the navigation pane on the left, choose **Clusters** > **Active Clusters**.
2. Click the name of a cluster to go to the cluster details page.
3. On the **Dashboard** page, click **Synchronize** on the right of **IAM User Sync** to synchronize IAM users.
4. Click **Manage Agency** on the right of **Agency**, select the target agency, and click **OK**.

Step 2 Create an OBS file system for storing data.

1. Log in to the OBS console.
2. In the navigation pane on the left, choose **Parallel File Systems**. On the displayed page, click **Create Parallel File System**.
3. Enter the file system name, for example, **esdk-c-test-pfs1**, and set other parameters as required. Click **Create Now**.
4. In the parallel file system list on the OBS console, click the created file system name to go to its details page.
5. In the navigation pane on the left, choose **Files** and click **Create Folder** to create the **testFlumeOutput** folder.

Step 3 Prepare the **properties.properties** file and upload it to the **/opt/flumeInput** directory.

1. Prepare the **properties.properties** file on the local host. Its content is as follows:

```
# source
server.sources = r1
# channels
server.channels = c1
# sink
server.sinks = obs_sink
# ----- define net source -----
server.sources.r1.type = seq
server.sources.r1.spooldir = /opt/flumeInput
# ---- define OBS sink ----
server.sinks.obs_sink.type = hdfs
server.sinks.obs_sink.hdfs.path = obs://esdk-c-test-pfs1/testFlumeOutput
server.sinks.obs_sink.hdfs.filePrefix = %[localhost]
server.sinks.obs_sink.hdfs.useLocalTimeStamp = true
# set file size to trigger roll
server.sinks.obs_sink.hdfs.rollSize = 0
server.sinks.obs_sink.hdfs.rollCount = 0
server.sinks.obs_sink.hdfs.rollInterval = 5
#server.sinks.obs_sink.hdfs.threadsPoolSize = 30
server.sinks.obs_sink.hdfs.fileType = DataStream
server.sinks.obs_sink.hdfs.writeFormat = Text
server.sinks.obs_sink.hdfs.fileCloseByEndEvent = false
```

```
# define channel
server.channels.c1.type = memory
server.channels.c1.capacity = 1000
# transaction size
server.channels.c1.transactionCapacity = 1000
server.channels.c1.byteCapacity = 800000
server.channels.c1.byteCapacityBufferPercentage = 20
server.channels.c1.keep-alive = 60
server.sources.r1.channels = c1
server.sinks.obs_sink.channel = c1
```

NOTE

The value of **server.sinks.obs_sink.hdfs.path** is the OBS file system created in **Step 2**.

2. Log in to the node where the Flume client is installed as user **root**.
3. Create the **/opt/flumeInput** directory and create a customized **.txt** file in this directory.
4. Log in to FusionInsight Manager.
5. Choose **Cluster** > *Name of the target cluster* > **Services** > **Flume**. On the displayed page, click **Configurations** and then **Upload File** in the **Value** column corresponding to the **flume.config.file** parameter, upload the **properties.properties** file prepared in **Step 3.1**, and click **Save**.

Step 4 View the result in the OBS system.

1. Log in to the OBS console.
2. Click **Parallel File Systems** and go to the folder created in **Step 2** to view the result.

----End

7.4.3 Interconnecting HDFS with OBS

Before performing the following operations, ensure that you have configured a storage-compute decoupled cluster by referring to **Configuring a Storage-Compute Decoupled Cluster (Agency)** or **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

Step 1 Log in to the node on which the HDFS client is installed as a client installation user.

Step 2 Run the following command to switch to the client installation directory.

cd ${client_home}

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, skip user authentication.

kinit *Component service user*

Step 5 Explicitly add the OBS file system to be accessed in the HDFS command line.

Example:

- Run the following command to access the OBS file system:

  hdfs dfs -ls obs://*OBS parallel file system name*/*Path*

- Run the following command to upload the **/opt/test.txt** file from the client node to the **/tmp** directory of HDFS:

  hdfs dfs -put /opt/test.txt /tmp

----End

NOTE

If a large number of logs are printed in the OBS file system, the read and write performance may be affected. You can adjust the log level of the OBS client as follows:

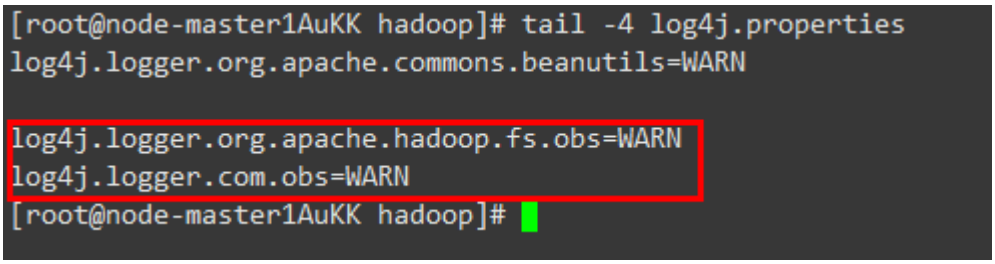
cd ${client_home}/HDFS/hadoop/etc/hadoop

vi log4j.properties

Add the OBS log level configuration to the file as follows:

log4j.logger.org.apache.hadoop.fs.obs=WARN

log4j.logger.com.obs=WARN



7.4.4 Interconnecting Hive with OBS

Before performing the following operations, ensure that you have configured a storage-compute decoupled cluster by referring to **Configuring a Storage-Compute Decoupled Cluster (Agency)** or **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

When creating a table, set the table location to an OBS path.

Step 1 Log in to the client installation node as the client installation user.

Step 2 Run the following command to initialize environment variables:

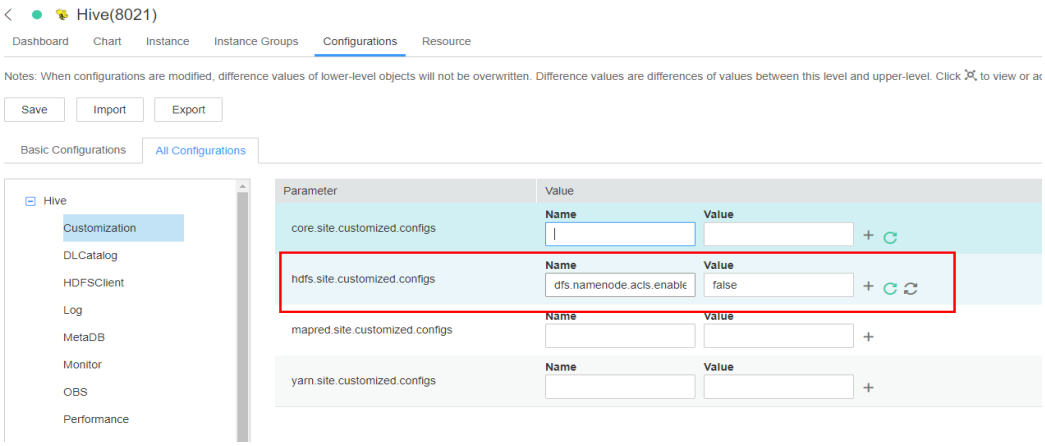
source ${client_home}/bigdata_env

Step 3 For a security cluster, run the following command to perform user authentication (the user must have the permission to perform Hive operations). If Kerberos authentication is not enabled for the current cluster, you do not need to run this command.

kinit *User performing Hive operations*

Step 4 Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Hive** > **Configurations** > **All Configurations**.

In the left navigation tree, choose **Hive** > **Customization**. In the customized configuration items, add **dfs.namenode.acls.enabled** to the **hdfs.site.customized.configs** parameter and set its value to **false**.



Step 5 Click **Save**. Click the **Dashboard** tab and choose **More** > **Restart Service**. In the **Verify Identity** dialog box that is displayed, enter the password of the current user, and click **OK**. In the displayed **Restart Service** dialog box, select **Restart upper-layer services** and click **OK**. Hive is restarted.

Step 6 Log in to the beeline client and set **Location** to the OBS file system path when creating a table.

beeline

For example, run the following command to create table **test** in **obs://***OBS parallel file system name/user/hive/warehouse/*:

create table test(name string) location "obs://*OBS parallel file system name/user/hive/warehouse/***";**

NOTE

You need to add the component operator to the URL policy in the Ranger policy. Set the URL to the complete path of the object on OBS. Select the Read and Write permissions.

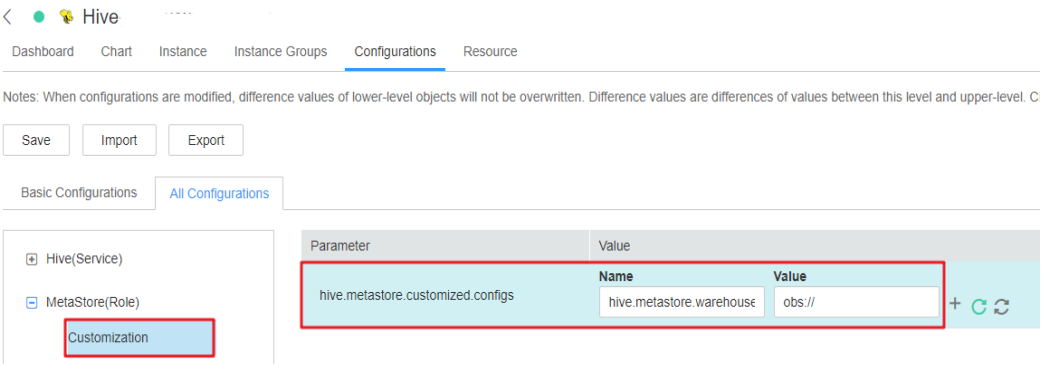
----End

Setting the Default Location of the Created Hive Table to the OBS Path

Step 1 Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Hive** > **Configurations** > **All Configurations**.

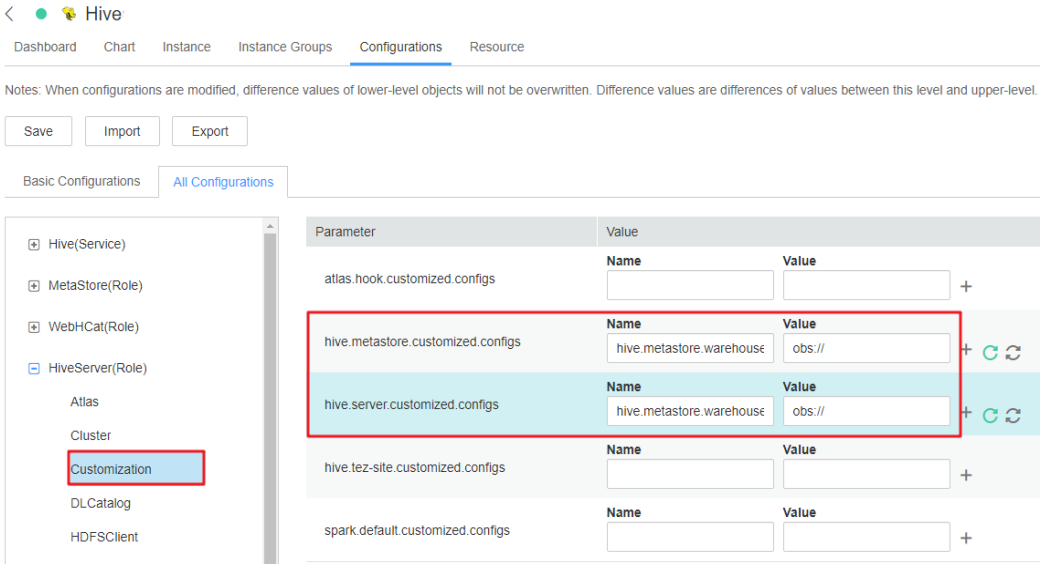
Step 2 In the left navigation tree, choose **MetaStore** > **Customization**. Add **hive.metastore.warehouse.dir** to the **hive.metastore.customized.configs** parameter and set it to the OBS path.

Figure 7-11 Configurations of **hive.metastore.warehouse.dir**



Step 3 In the left navigation tree, choose **HiveServer** > **Customization**. Add **hive.metastore.warehouse.dir** to the **hive.metastore.customized.configs** and **hive.metastore.customized.configs** parameters, and set it to the OBS path.

Figure 7-12 hive.metastore.warehouse.dir configuration



Step 4 Save the configurations and restart Hive.

Step 5 Update the client configuration file.

1. Run the following command to open **hivemetastore-site.xml** in the Hive configuration file directory on the client:

   vim /opt/Bigdata/client/Hive/config/hivemetastore-site.xml

2. Change the value of **hive.metastore.warehouse.dir** to the corresponding OBS path.

```
</property>
<property>
<name>hive.metastore.warehouse.dir</name>
<value>obs://</value>
</property>
<property>
<name>hive.metastore.metrics.enabled</name>
```

Step 6 Log in to the beeline client, create a table, and check whether the location is the OBS path.

beeline

create table test(name string);

desc formatted test;

NOTE

If the database location points to HDFS, the table to be created in the database (without specifying the location) also points to HDFS. If you want to modify the default table creation policy, change the location of the database to OBS by performing the following operations:

1. Run the following command to query the location of the database:

   show create database *obs_test*;

   ```
   +----------------------------------------------------+
   |                    createdb_stmt                   |
   +----------------------------------------------------+
   | CREATE DATABASE `obs_test`                         |
   | LOCATION                                           |
   |   'hdfs://hacluster/user/hive/warehouse/obs_test.db'  |
   +----------------------------------------------------+
   3 rows selected (0.038 seconds)
   ```

2. Run the following command to change the database location:

   alter database *obs_test* **set location** *'obs://test1231/'*

   Run the **show create database** *obs_test* command to check whether the database location points to OBS.

   ```
   INFO  : Concurrency mode is disabled, not creating
   +----------------------------+
   |        createdb_stmt       |
   +----------------------------+
   | CREATE DATABASE `obs_test` |
   | LOCATION                   |
   |   'obs://test1231/'        |
   +----------------------------+
   3 rows selected (0.063 seconds)
   ```

3. Run the following command to change the table location:

   alter table *user_info* **set location** *'obs://test1231/'*

   If the table contains data, migrate the original data file to the new location.

----End

7.4.5 Interconnecting MapReduce with OBS

Before performing the following operations, ensure that you have configured a storage-compute decoupled cluster by referring to **Configuring a Storage-Compute Decoupled Cluster (Agency)** or **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

Step 1 Log in to the MRS management console and click the cluster name to go to the cluster details page.

Step 2 Choose **Components > MapReduce**. The **All Configurations** page is displayed. In the navigation tree on the left, choose **MapReduce > Customization**. In the customized configuration items, add the configuration item **mapreduce.jobhistory.always-scan-user-dir** to **core-site.xml** and set its value to **true**.

Step 3 Save the configurations and restart the MapReduce service.

----End

7.4.6 Interconnecting Spark2x with OBS

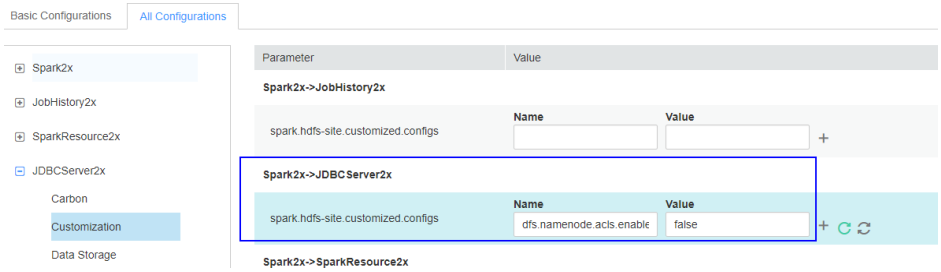
The OBS file system can be interconnected with Spark2x after an MRS cluster is installed.

Before performing the following operations, ensure that you have configured a storage-compute decoupled cluster by referring to **Configuring a Storage-Compute Decoupled Cluster (Agency)** or **Configuring a Storage-Compute Decoupled Cluster (AK/SK)**.

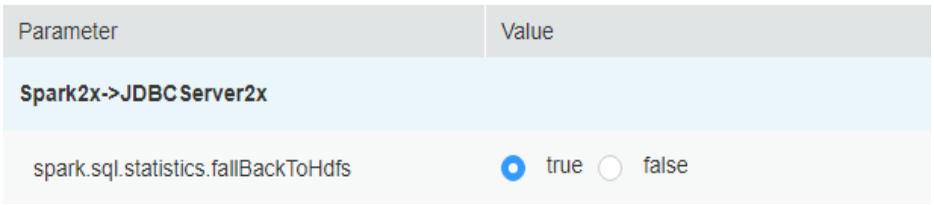
Using Spark Beeline After Cluster Installation

Step 1 Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Spark2x** > **Configurations** > **All Configurations**.

In the left navigation tree, choose **JDBCServer2x** > **Customization**. Add **dfs.namenode.acls.enabled** to the **spark.hdfs-site.customized.configs** parameter and set its value to **false**.



Step 2 Search for the **spark.sql.statistics.fallBackToHdfs** parameter and set its value to **false**.



Step 3 Save the configurations and restart the JDBCServer2x instance.

Step 4 Log in to the client installation node as the client installation user.

Step 5 Run the following command to configure environment variables:

source ${client_home}/bigdata_env

Step 6 For a security cluster, run the following command to perform user authentication. If Kerberos authentication is not enabled for the current cluster, you do not need to run this command.

kinit *Username*

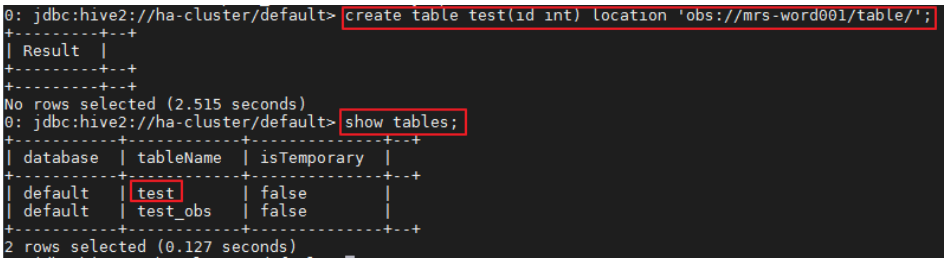
Step 7 Access OBS in spark-beeline. For example, create a table named **test** in the **obs://mrs-word001/table/** directory.

create table test(id int) location '*obs://mrs-word001/table/*';

Step 8 Run the following command to query all tables. If table **test** is displayed in the command output, OBS access is successful.

show tables;

Figure 7-13 Verifying the created table name returned using Spark2x



Step 9 Press **Ctrl+C** to exit the Spark Beeline.

----End

Using Spark SQL After Cluster Installation

Step 1 Log in to the client installation node as the client installation user.

Step 2 Run the following command to configure environment variables:

source ${client_home}/bigdata_env

Step 3 Modify the configuration file:

vim ${client_home}/Spark2x/spark/conf/hdfs-site.xml

```
<property>
<name>dfs.namenode.acls.enabled</name>
<value>false</value>
</property>
```

Step 4 For a security cluster, run the following command to perform user authentication. If Kerberos authentication is not enabled for the current cluster, you do not need to run this command.

kinit *Username*

Step 5 Access OBS in spark-sql. For example, create a table named **test** in the **obs://mrs-word001/table/** directory.

Step 6 Run the **cd ${client_home}/Spark2x/spark/bin** command to access the **spark bin** directory and run **./spark-sql** to log in to spark-sql CLI.

Step 7 Run the following command in the spark-sql CLI:

create table test(id int) location '*obs://mrs-word001/table/***';**

Step 8 Run the **show tables;** command to confirm that the table is created successfully.

Step 9 Run **exit;** to exit the spark-sql CLI.

NOTE

If a large number of logs are printed in the OBS file system, the read and write performance may be affected. You can adjust the log level of the OBS client as follows:

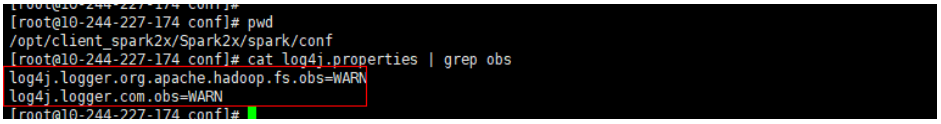
cd ${client_home}/Spark2x/spark/conf

vi log4j.properties

Add the OBS log level configuration to the file as follows:

log4j.logger.org.apache.hadoop.fs.obs=WARN

log4j.logger.com.obs=WARN



----End

7.4.7 Interconnecting Sqoop with External Storage Systems

Exporting Data From HDFS to MySQL Using the sqoop export Command

Step 1 Log in to the node where the client is located.

Step 2 Run the following command to initialize environment variables:

source /opt/client/bigdata_env

Step 3 Run the following command to operate the Sqoop client:

sqoop export --connect jdbc:mysql://10.100.231.134:3306/test --username root --password xxxxxx --table component13 -export-dir hdfs://hacluster/user/hive/warehouse/component_test3 --fields-terminated-by ',' -m 1

Table 7-1 Parameter description

Parameter	Description
-direct	Imports data to a relational database using a database import tool, for example, mysqlimport of MySQL, more efficient than the JDBC connection mode.
-export-dir <dir>	Specifies the source directory for storing data in the HDFS.
-m or -num-mappers <n>	Starts *n* (4 by default) maps to import data concurrently. The value cannot be greater than the maximum number of maps in a cluster.
-table <table-name>	Specifies the relational database table to be imported.

Parameter	Description
-update-key <col-name>	Specifies the column used for updating the existing data in a relational database.
-update-mode <mode>	Specifies how updates are performed. The value can be **updateonly** or **allowinsert**. This parameter is used only when the relational data table does not contain the data record to be imported. For example, if the HDFS data to be imported to the destination table contains a data record **id=1** and the table contains an existing data record **id=2**, the update will fail.
-input-null-string <null-string>	This parameter is optional. If it is not specified, **null** will be used.
-input-null-non-string <null-string>	This parameter is optional. If it is not specified, **null** will be used.
-staging-table <staging-table-name>	Creates a table with the same data structure as the destination table for storing data before it is imported to the destination table. This parameter ensures the transaction security when data is imported to a relational database table. Due to multiple transactions during an import, this parameter can prevent other transactions from being affected when one transaction fails. For example, the imported data is incorrect or duplicate records exist.
-clear-staging-table	Clears data in the staging table before data is imported if the staging-table is not empty.

----End

Importing Data from MySQL to Hive Using the sqoop import Command

Step 1 Log in to the node where the client is located.

Step 2 Run the following command to initialize environment variables:

source /opt/client/bigdata_env

Step 3 Run the following command to operate the Sqoop client:

sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password xxxxxx --table component --hive-import --hive-table component_test2 --delete-target-dir --fields-terminated-by "," -m 1 --as-textfile

Table 7-2 Parameter description

Parameter	Description
-append	Appends data to an existing dataset in the HDFS. Once this parameter is used, Sqoop imports data to a temporary directory, renames the temporary file where the data is stored, and moves the file to a formal directory to avoid duplicate file names in the directory.
-as-avrodatafile	Imports data to a data file in the Avro format.
-as-sequencefile	Imports data to a sequence file.
-as-textfile	Import data to a text file. After the text file is generated, you can run SQL statements in Hive to query the result.
-boundary-query <statement>	Specifies the SQL statement for performing boundary query. Before importing data, use a SQL statement to obtain a result set and import the data in the result set. The data format can be **-boundary-query 'select id,creationdate from person where id = 3'** (indicating a data record whose ID is 3) or **select min(<split-by>), max(<split-by>) from <table name>**. The fields to be queried cannot contain fields whose data type is string. Otherwise, the error message "java.sql.SQLException: Invalid value for getLong()" is displayed.
-columns<col,col,col...>	Specifies the fields to be imported. The format is ***-Column id,Username***.
-direct	Imports data to a relational database using a database import tool, for example, mysqlimport of MySQL, more efficient than the JDBC connection mode.
-direct-split-size	Splits the imported streams by byte. Especially when data is imported from PostgreSQL using the direct mode, a file that reaches the specified size can be divided into several independent files.
-inline-lob-limit	Sets the maximum value of an inline LOB.
-m or -num-mappers	Starts *n* (4 by default) maps to import data concurrently. The value cannot be greater than the maximum number of maps in a cluster.
-query, -e<statement>	Imports data from the query result. To use this parameter, you must specify the **-target-dir** and **-hive-table** parameters and use the query statement containing the WHERE clause as well as $CONDITIONS. Example: **-query'select * from person where $CONDITIONS' -target-dir /user/hive/warehouse/person -hive-table person**

Parameter	Description
-split-by<column-name>	Specifies the column of a table used to split work units. Generally, the column name is followed by the primary key ID.
-table <table-name>	Specifies the relational database table from which data is obtained.
-target-dir <dir>	Specifies the HDFS path.
-warehouse-dir <dir>	Specifies the directory for storing data to be imported. This parameter is applicable when data is imported to HDFS but cannot be used when you import data to Hive directories. This parameter cannot be used together with **-target-dir**.
-where	Specifies the WHERE clause when data is imported from a relational database, for example, **-where 'id = 2'**.
-z,-compress	Compresses sequence, text, and Avro data files using the GZIP compression algorithm. Data is not compressed by default.
–compression-codec	Specifies the Hadoop compression codec. GZIP is used by default.
–null-string <null-string>	Specifies the string to be interpreted as **NULL** for string columns.
–null-non-string<null-string>	Specifies the string to be interpreted as null for non-string columns. If this parameter is not specified, **NULL** will be used.
-check-column (col)	Specifies the column for checking incremental data import, for example, **id**.
-incremental (mode) append or last modified	Incrementally imports data. **append**: appends records, for example, appending records that are greater than the value specified by **last-value**. **lastmodified**: appends data that is modified after the date specified by **last-value**.
-last-value (value)	Specifies the maximum value (greater than the specified value) of the column after the last import. This parameter can be set as required.

----End

Sqoop Usage Example

- Importing data from MySQL to HDFS using the **sqoop import** command

sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* --query 'SELECT * FROM component where $CONDITIONS and component_id ="MRS 1.0_002"' --target-dir /tmp/component_test --delete-target-dir --fields-terminated-by "," -m 1 --as-textfile

- Exporting data from OBS to MySQL using the **sqoop export** command

  sqoop export --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* --table component14 -export-dir obs://obs-file-bucket/xx/part-m-00000 --fields-terminated-by ',' -m 1

- Importing data from MySQL to OBS using the **sqoop import** command

  sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* --table component --target-dir obs://obs-file-bucket/xx --delete-target-dir --fields-terminated-by "," -m 1 --as-textfile

- Importing data from MySQL to OBS tables outside Hive

  sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* --table component --hive-import --hive-table component_test01 --fields-terminated-by "," -m 1 --as-textfile

8 Accessing Web Pages of Open Source Components Managed in MRS Clusters

8.1 Web UIs of Open Source Components

Scenario

Web UIs of different components are created and hosted on the Master or Core nodes in the MRS cluster by default. You can view information about the components on these web UIs.

Procedure for accessing the web UIs of open-source component:

1. Select an access method.

   MRS provides the following methods for accessing the web UIs of open-source components:

   - **EIP-based Access**: This method is recommended because it is easy to bind an EIP to a cluster.
   - **Access Using a Windows ECS**: Independent ECSs need to be created and configured.
   - **Creating an SSH Channel for Connecting to an MRS Cluster and Configuring the Browser**: Use this method when the user and the MRS cluster are on different networks.

2. Access the web UIs. For details, see **Table 8-1**.

Web UIs

NOTE

For clusters with Kerberos authentication enabled, user **admin** does not have the management permission on each component. To access the web UI of each component, add a user who has the management permission on the corresponding component.

Table 8-1 Web UI addresses of open-source components

Cluster Type	Web UI Type	Web UI Address
All Types	MRS Manager	• Applicable to clusters of all versions **https://***Floating IP address of Manager***:28443/web** **NOTE** 1. Ensure that the local host can communicate with the MRS cluster. 2. Log in to the Master2 node remotely, and run the **ifconfig** command. In the command output, **eth0:wsom** indicates the floating IP address of MRS Manager. Record the value of **inet**. If the floating IP address of MRS Manager cannot be queried on the Master2 node, switch to the Master1 node to query and record the floating IP address. If there is only one Master node, query and record the cluster manager IP address of the Master node. • For versions earlier than MRS 3.x: https://<*EIP*>:9022/mrsmanager?locale=en-us For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. • For MRS 3.*x* or later, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.
Analysis cluster	HDFS NameNode	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **HDFS** > **NameNode Web UI** > **NameNode (Active)**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **HDFS** > **NameNode Web UI** > **NameNode (*Host name*, Active)**.

Cluster Type	Web UI Type	Web UI Address
	HBase HMaster	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **HBase** > **HMaster Web UI** > **HMaster (Active)**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **HBase** > **HMaster Web UI** > **HMaster (*Host name*, Active)**.
	MapReduce JobHistoryServer	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **MapReduce** > **JobHistoryServer Web UI** > **JobHistoryServer**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **MapReduce** > **JobHistoryServer Web UI** > **JobHistoryServer (*Host name*, Active)**.
	YARN ResourceManager	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Yarn** > **ResourceManager Web UI** > **ResourceManager (Active)**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **Yarn** > **ResourceManager Web UI** > **ResourceManager (*Host name*, Active)**.
	Spark JobHistory	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Spark** > **Spark Web UI** > **JobHistory**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **Spark2x** > **Spark2x Web UI** > **JobHistory2x (*Host name*, Active)**.

Cluster Type	Web UI Type	Web UI Address
	Hue	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Hue** > **Hue Web UI** > **Hue (Active)**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **Hue** > **Hue Web UI** > **Hue (*Host name*, Active)**. Loader is a graphical data migration management tool based on the open-source Sqoop web UI, and its interface is hosted on the Hue web UI.
	Tez	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Tez** > **Tez Web UI** > **TezUI**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **Tez** > **Tez Web UI** > **TezUI (*Host name*, Active)**.
	Presto	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Presto** > **Presto Web UI** > **Coordinator (Active)**. • On the Manager homepage, choose **Cluster** > **Services** > **Presto** > **Coordinator Web UI** > **Coordinator (Coordinator)**.
	Ranger	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Ranger** > **Ranger Web UI** > **RangerAdmin (Active)**. • MRS 3.x or later: On the Manager homepage, choose **Cluster** > **Services** > **Ranger** > **Ranger Web UI** > **RangerAdmin**.
Stream processing cluster	Storm	• Versions earlier than MRS 3.x: On the cluster details page, choose **Components** > **Storm** > **Storm Web UI** > **UI**. • On the Manager homepage, choose **Cluster** > **Services** > **Storm** > **Storm Web UI** > **UI (*Host name*)**.

8.2 List of Open Source Component Ports

Common HBase Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
hbase.master.port	16000	HMaster RPC port. This port is used to connect the HBase client to HMaster. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes
hbase.master.info.port	16010	HMaster HTTPS port. This port is used by the remote web client to connect to the HMaster UI. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes
hbase.regionserver.port	16020	RegoinServer (RS) RPC port. This port is used to connect the HBase client to RegionServer. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
hbase.regionserver.info.port	16030	HTTPS port of the Region server. This port is used by the remote web client to connect to the RegionServer UI. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
hbase.thrift.info.port	9095	Thrift Server listening port of Thrift Server. This port is used for: Listening when the client is connected **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
hbase.regionserver.thrift.port	9090	Thrift Server listening port of RegionServer. This port is used for: Listening when the client is connected to the RegionServer **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
hbase.rest.info.port	8085	Port of the RegionServer RESTServer native web page
-	21309	REST port of RegionServer RESTServer

Common HDFS Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
dfs.namenode.rpc.port	• 9820 (versions earlier than MRS 3.*x*) • 8020 (MRS 3.*x* and later)	NameNode RPC port. This port is used for: 1. Communication between the HDFS client and NameNode 2. Connection between the DataNode and NameNode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes
dfs.namenode.http.port	9870	HDFS HTTP port (NameNode). This port is used for: 1. Point-to-point NameNode checkpoint operations. 2. Connecting the remote web client to the NameNode UI **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes
dfs.namenode.https.port	9871	HDFS HTTPS port (NameNode). This port is used for: 1. Point-to-point NameNode checkpoint operations 2. Connecting the remote web client to the NameNode UI **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
dfs.datanode.ipc.port	9867	IPC server port of DataNode. This port is used for: Connection between the client and DataNode to perform RPC operations. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.datanode.port	9866	DataNode data transmission port. This port is used for: 1. Transmitting data from HDFS client from or to the DataNode 2. Point-to-point DataNode data transmission **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.datanode.http.port	9864	DataNode HTTP port. This port is used for: Connecting to the DataNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
dfs.datanode.https.port	9865	HTTPS port of DataNode. This port is used for: Connecting to the DataNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.JournalNode.rpc.port	8485	RPC port of JournalNode. This port is used for: Client communication to access multiple types of information **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.journalnode.http.port	8480	JournalNode HTTP port This port is used for: Connecting to the JournalNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
dfs.journalnode.https.port	8481	HTTPS port of JournalNode. This port is used for: Connecting to the JournalNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
httpfs.http.port	14000	Listening port of the HttpFS HTTP server. This port is used for: Connecting to the HttpFS from the remote REST API **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Common Hive Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
templeton.port	9111	Port used by WebHCat for providing the REST service. This port is used for: Communication between the WebHCat client and WebHCat server ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
hive.server2.thrift.port	10000	Port for HiveServer to provide Thrift services. This port is used for: Communication between the HiveServer and HiveServer client ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
hive.metastore.port	9083	Port for MetaStore to provide Thrift services. This port is used for: Communication between the MetaStore client and MetaStore, that is, communication between HiveServer and MetaStore. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
hive.server2.webui.port	10002	Web UI port of Hive. This port is used for: HTTPS/HTTP communication between Web requests and the Hive UI server

Common Hue Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
HTTP_PORT	8888	Port for Hue to provide HTTPS services. This port is used for: providing web services in HTTPS mode (The port can be modified.) ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Common Kafka Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
port	9092	Port for a broker to receive data and obtain services
ssl.port	9093	SSL port used by a broker to receive data and obtain services
sasl.port	21007	SASL security authentication port provided by a broker, which provides the secure Kafka service
sasl-ssl.port	21009	Port used by a broker to provide encrypted service based on the SASL and SSL protocols

Common Loader Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
LOADER_HTTPS_PORT	21351	This port is used for: providing REST APIs for configuration and running of Loader jobs • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Common Manager Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
-	8080	Port provided by WebService for user access. This port is used to access the web UI over HTTP. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
-	28443	Port provided by WebService for user access. This port is used to access the web UI over HTTPS. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Common MapReduce Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
mapreduce.jobhistory.webapp.port	19888	Web HTTP port of the JobHistory server. This port is used for: viewing the web page of the JobHistory server **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
mapreduce.jobhistory.port	10020	Port of the JobHistory server. This port is used for: 1. Task data restoration in the MapReduce client 2. Obtaining task report in the Job client **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
mapreduce.jobhistory.webapp.https.port	19890	Web HTTPS port of the JobHistory server. This port is used for: viewing the web page of the JobHistory server **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Common Spark Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
hive.server2.thrift.port	22550	JDBC thrift port. This port is used for: Socket communication between Spark2.1.0 CLI/ JDBC client and server **NOTE** If **hive.server2.thrift.port** is occupied, an exception indicating that the port is occupied is reported. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
spark.ui.port	4040	Web UI port of JDBC This port is used for: HTTPS/HTTP communication between Web requests and the JDBC Server Web UI server **NOTE** The system verifies the port configuration. If the port is invalid, the value of the port plus 1 is used till the calculated value is valid. (A maximum number of 16 attempts are allowed. The number of attempts is specified by **spark.port.maxRetries**.) • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes
spark.history.ui.port	18080	JobHistory Web UI port This port is used for: HTTPS/HTTP communication between Web requests and Spark2.1.0 History Server **NOTE** The system verifies the port configuration. If the port is invalid, the value of the port plus 1 is used till the calculated value is valid. (A maximum number of 16 attempts are allowed. The number of attempts is specified by **spark.port.maxRetries**.) • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Common Storm Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
nimbus.thrift.port	6627	Port for Nimbus to provide thrift services

Parameter	Default Port	Port Description
supervisor.slots.ports	6700,6701,6702,6703	Port for receiving service requests that are forwarded from other servers
logviewer.https.port	29248	Port for the LogViewer to provide HTTPS services
ui.https.port	29243	Port for the Storm UI to provide HTTPS services (ui.https.port)

Common Yarn Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
yarn.resourcemanager.webapp.port	8088	Web HTTP port of the ResourceManager service.
yarn.resourcemanager.webapp.https.port	8090	Web HTTPS port of the ResourceManager service. This port is used for: accessing the Resource Manager web applications in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
yarn.nodemanager.webapp.port	8042	NodeManager Web HTTP port
yarn.nodemanager.webapp.https.port	8044	NodeManager Web HTTPS port. This port is used for: Accessing the NodeManager web application in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Common ZooKeeper Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
clientPort	2181	ZooKeeper client port. This port is used for: Connection between the ZooKeeper client and server. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Common Kerberos Ports

The protocol type of all ports in the table is UDP.

Parameter	Default Port	Port Description
kdc_ports	21732	Kerberos server port. This port is used for: Performing Kerberos authentication for components This parameter may be used during the configuration of mutual trust between clusters. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Common OpenTSDB Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
tsd.network.port	4242	Web UI port of OpenTSDB. This port is used for: HTTPS/HTTP communication between web requests and the OpenTSDB UI server

Common Tez Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
tez.ui.port	28888	Web UI port of Tez

Common KafkaManager Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
kafka_manager_port	9099	Web UI port of KafkaManager.

Common Presto Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
http-server.http.port	7520	HTTP port of Presto coordinator to provide services for external entities
http-server.https.port	7521	HTTPS port of Presto coordinator to provide services for external entities
http-server.http.port	7530	HTTP port used by Presto worker to provide services for external entities
http-server.https.port	7531	HTTPS port used by Presto worker to provide services for external entities

Common Flink Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
jobmanager.web.port	32261-32325	Web UI port of Flink. This port is used for: HTTP/HTTPS communication between the client web requests and Flink server

Common ClickHouse Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
tcp_port	9000	TCP port for accessing the service client.
http_port	8123	HTTP port for accessing the service client.
https_port	8443	HTTPS port for accessing the service client.
tcp_port_secure	9440	TCP With SSL port for accessing the service client. This port is enabled only in security mode by default.

Common Impala Ports

The protocol type of the port in the table is TCP.

Parameter	Default Port	Port Description
--beeswax_port	21000	Port for impala-shell communication.
--hs2_port	21050	Port for Impala application communication.
--hs2_http_port	28000	Port used by Impala to provide the HiveServer2 protocol for external systems.

8.3 Access Through Direct Connect

MRS allows you to access MRS clusters using Direct Connect. Direct Connect is a high-speed, low-latency, stable, and secure dedicated network connection that connects your local data center to an online cloud VPC. It extends online cloud

services and existing IT facilities to build a flexible, scalable hybrid cloud computing environment.

Prerequisites

Direct Connect is available, and the connection between the local data center and the online VPC has been established.

Accessing an MRS Cluster Using Direct Connect

Step 1 Log in to the MRS console.

Step 2 Click the name of the cluster to enter its details page.

Step 3 On the **Dashboard** tab page of the cluster details page, click **Access Manager** next to **MRS Manager**.

Step 4 Set **Access Mode** to **Direct Connect** and select **I confirm that the network between the local PC and the floating IP address is connected and that MRS Manager is accessible using the Direct Connect connection**.

The floating IP address is automatically allocated by MRS to access MRS Manager. Before using Direct Connect to access MRS Manager, ensure that the connection between the local data center and the online VPC has been established.

Step 5 Click **OK**. The MRS Manager login page is displayed. Enter the username **admin** and the password set during cluster creation.

----End

Switching the MRS Manager Access Mode

To facilitate user operations, the browser cache records the selected Manager access mode. To change the access mode, perform the following steps:

Step 1 Log in to the MRS console.

Step 2 Click the name of the cluster to enter its details page.

Step 3 On the **Dashboard** tab page of the cluster details page, click ⇌ next to **MRS Manager**.

Step 4 On the displayed page, set **Access Mode**.

- To change **EIP** to **Direct Connect**, ensure that the network for direct connections is available, set **Access Mode** to **Direct Connect**, and select **I confirm that the network between the local PC and the floating IP address is connected and that MRS Manager is accessible using the Direct Connect connection**. Click **OK**.
- To change **Direct Connect** to **EIP**, set **Access Mode** to **EIP** and configure the EIP by referring to **Accessing FusionInsight Manager Using an EIP**. If a public IP address has been configured for the cluster, click **OK** to access MRS Manager using an EIP.

----End

8.4 EIP-based Access

You can bind an EIP to a cluster to access the web UIs of the open-source components managed in the MRS cluster. This method is simple and easy to use and is recommended for accessing the web UIs of the open-source components.

Binding an EIP to a Cluster and Adding a Security Group Rule

1. On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users. After the IAM users are synchronized, the **Components** tab is available.
2. Click **Access Manager** on the right of **MRS Manager**.
3. The page for accessing MRS Manager is displayed. Bind an EIP and add a security group rule. Perform the following operations only when you access the web UIs of the open-source components of the cluster for the first time.
   a. Select an available EIP from the EIP drop-down list to bind it. If there is no available EIP, click **Manage EIP** to an EIP. If an EIP has been bound during cluster creation, skip this step.
   b. Select the security group to which the security group rule to be added belongs. The security group is configured when the group is created.
   c. Add a security group rule. By default, your public IP address used for accessing port 9022 is filled in the rule. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

      NOTE

      - It is normal that the automatically generated public IP address is different from the local IP address and no action is required.
      - If port 9022 is a Knox port, you need to enable the permission of port 9022 to access Knox for accessing MRS components.

   d. Select the checkbox stating that **I confirm that xx.xx.xx.xx is a trusted public IP address and MRS Manager can be accessed using this IP address**.
   e. Click **OK**. The login page is displayed. Enter the username **admin** and the password set during cluster creation.
4. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **HDFS**. On the displayed page, click **NameNode(*Host name*, active)** to access the HDFS web UI. The HDFS NameNode is used as an example. For details about the web UIs of other components, see **Web UIs of Open Source Components**.

8.5 Access Using a Windows ECS

MRS allows you to access the web UIs of open-source components through a Windows ECS. This method is complex and is recommended for MRS clusters that do not support the EIP function.

Step 1 On the MRS management console, click **Clusters**.

Step 2 On the **Active Clusters** page, click the name of the specified cluster.

On the cluster details page, record the **AZ**, **VPC**, **Cluster Manager IP Address**, and **Security Group** of the cluster.

NOTE

To obtain the cluster manager IP address, remotely log in to the Master2 node, and run the **ifconfig** command. In the command output, **eth0:wsom** indicates the cluster manager IP address. Record the value of **inet**. If the cluster manager IP address cannot be queried on the Master2 node, switch to the Master1 node to query and record the cluster manager IP address. If there is only one Master node, query and record the cluster manager IP address of the Master node.

Step 3 On the ECS management console, create an ECS.

- The **AZ**, **VPC**, and **Security Group** of the ECS must be the same as those of the cluster to be accessed.
- Select a Windows public image. For example, select the standard image **Windows Server 2012 R2 Standard 64bit(40GB)**.
- For details about other configuration parameters, see **Elastic Cloud Server > User Guide > Getting Started > Creating and Logging In to a Windows ECS**.

NOTE

If the security group of the ECS is different from **Security Group** of the MRS cluster, you can modify the configuration using either of the following methods:

- Change the security group of the ECS to the security group of the MRS cluster. For details, see **Elastic Cloud Server** > **User Guide** > **Security Group** > **Changing a Security Group**.
- Add two security group rules to the security groups of the Master and Core nodes to enable the ECS to access the cluster. Set **Protocol** to **TCP** and **ports** of the two security group rules to **28443** and **20009**, respectively. For details, see **Virtual Private Cloud** > **User Guide** > **Security** > **Security Group** > **Adding a Security Group Rule**.

Step 4 On the VPC management console, apply for an EIP and bind it to the ECS.

For details, see **Virtual Private Cloud** > **User Guide** > **Elastic IP** > **Assigning an EIP and Binding It to an ECS**.

Step 5 Log in to the ECS.

The Windows system account, password, EIP, and the security group rules are required for logging in to the ECS. For details, see **Elastic Cloud Server > User Guide > Instances > Logging In to a Windows ECS**.

Step 6 On the Windows remote desktop, use your browser to access Manager.

For example, you can use Internet Explorer 11 in the Windows 2012 OS.

The MRS Manager access address is in the format of **https://*Cluster Manager IP Address*:28443/web**. Enter the name and password of the MRS cluster user, for example, user **admin**.

NOTE

- To obtain the cluster manager IP address, remotely log in to the Master2 node, and run the **ifconfig** command. In the command output, **eth0:wsom** indicates the cluster manager IP address. Record the value of **inet**. If the cluster manager IP address cannot be queried on the Master2 node, switch to the Master1 node to query and record the cluster manager IP address. If there is only one Master node, query and record the cluster manager IP address of the Master node.
- If you access MRS Manager with other MRS cluster usernames, change the password upon your first access. The new password must meet the requirements of the current password complexity policies.
- By default, a user is locked after inputting an incorrect password five consecutive times. The user is automatically unlocked after 5 minutes.

Step 7 Visit the web UIs of the open-source components by referring to the addresses listed in **Web UIs of Open Source Components**.

----End

Related Tasks

Configuring the Mapping Between Cluster Node Names and IP Addresses

Step 1 Log in to MRS Manager, and choose **Host Management**.

Record the host names and management IP addresses of all nodes in the cluster.

Step 2 In the work environment, use Notepad to open the **hosts** file and add the mapping between node names and IP addresses to the file.

Fill in one row for each mapping relationship, as shown in the following figure.

```
192.168.4.127 node-core-Jh3ER
192.168.4.225 node-master2-PaWVE
192.168.4.19 node-core-mtZ81
192.168.4.33 node-master1-zbYN8
192.168.4.233 node-core-7KoGY
```

Save the modifications.

----End

8.6 Creating an SSH Channel for Connecting to an MRS Cluster and Configuring the Browser

Scenario

Users and an MRS cluster are in different networks. As a result, an SSH channel needs to be created to send users' requests for accessing websites to the MRS cluster and dynamically forward them to the target websites.

The MAC system does not support this function. For details about how to access MRS, see **EIP-based Access**.

Prerequisites

- You have prepared an SSH client for creating the SSH channel, for example, the Git open-source SSH client. You have downloaded and installed the client.
- You have created a cluster and prepared a key file in PEM format or obtained the password used during cluster creation.
- Users can access the Internet on the local PC.

Procedure

Step 1 Log in to the MRS management console and choose **Clusters** > **Active Clusters**.

Step 2 Click the specified MRS cluster name.

Record the security group of the cluster.

Step 3 Add an inbound rule to the security group of the Master node to allow data access to the IP address of the MRS cluster through port **22**.

For details, see **Virtual Private Cloud** > **User Guide** > **Security** > **Security Group** > **Adding a Security Group Rule**.

Step 4 Query the primary management node of the cluster. For details, see **Determining Active and Standby Management Nodes of Manager**.

Step 5 Bind an elastic IP address to the primary management node.

For details, see **Virtual Private Cloud** > **User Guide** > **Elastic IP** > **Assigning an EIP and Binding It to an ECS**.

Step 6 Start Git Bash locally and run the following command to log in to the active management node of the cluster: **ssh root@*Elastic IP address*** or **ssh -i *Path of the key file* root@*Elastic IP address***.

Step 7 Run the following command to view data forwarding configurations:

cat /etc/sysctl.conf | grep net.ipv4.ip_forward

- If **net.ipv4.ip_forward=1** is displayed, the forwarding function has been configured. Go to **Step 9**.
- If **net.ipv4.ip_forward=0** is displayed, the forwarding function has not been configured. Go to **Step 8**.
- If **net.ipv4.ip_forward** fails to be queried, this parameter has not been configured. Run the following command and then go to **Step 9**:

  echo "net.ipv4.ip_forward = 1" >> /etc/sysctl.conf

Step 8 Modify forwarding configurations on the node.

1. Run the following command to switch to user **root**:

   sudo su - root

2. Run the following commands to modify forwarding configurations:

   echo 1 > /proc/sys/net/ipv4/ip_forward

   sed -i "s/net.ipv4.ip_forward=0/net.ipv4.ip_forward = 1/g" /etc/sysctl.conf

   sysctl -w net.ipv4.ip_forward=1

3. Run the following command to modify the **sshd** configuration file:

vi /etc/ssh/sshd_config

Press **I** to enter the edit mode. Locate **AllowTcpForwarding** and **GatewayPorts** and delete comment tags. Modify them as follows. Save the changes and exit.

```
AllowTcpForwarding yes
GatewayPorts yes
```

4. Run the following command to restart the sshd service:

   service sshd restart

Step 9 Run the following command to view the floating IP address:

ifconfig

In the command output, **eth0:FI_HUE** indicates the floating IP address of Hue and **eth0:wsom** specifies the floating IP address of Manager. Record the value of **inet**.

Run the **exit** command to exit.

Step 10 Run the following command on the local PC to create an SSH channel supporting dynamic port forwarding:

ssh -i *Path of the key file* -v -ND *Local port* root@*Elastic IP address* or ***ssh -v -ND Local port root@Elastic IP address***. After running the command, enter the password you set when you create the cluster.

In the command, set **Local port** to the user's local port that is not occupied. Port **8157** is recommended.

After the SSH channel is created, add **-D** to the command and run the command to start the dynamic port forwarding function. By default, the dynamic port forwarding function enables a SOCKS proxy process and monitors the user's local port. Port data will be forwarded to the primary management node using the SSH channel.

Step 11 Run the following command to configure the browser proxy.

1. Go to the Google Chrome client installation directory on the local PC.
2. Press **Shift** and right-click the blank area, choose **Open Command Window Here** and enter the following command:

   chrome --proxy-server="socks5://localhost:8157" --host-resolver-rules="MAP * 0.0.0.0 , EXCLUDE localhost" --user-data-dir=c:/tmppath --proxy-bypass-list="*google*com,*gstatic.com,*gvt*.com,*:80"

   NOTE

   - In the preceding command, **8157** is the local proxy port configured in **Step 10**.
   - If the local OS is Windows 10, start the Windows OS, click **Start** and enter **cmd**. In the displayed CLI, run the command in **Step 11.2**. If this method fails, click **Start**, enter the command in the search box, and run the command in **Step 11.2**.

Step 12 In the address box of the browser, enter the address for accessing Manager.

Address format: **https://***Floating IP address of FusionInsight Manager***:28443/web**

The username and password of the MRS cluster need to be entered for accessing clusters with Kerberos authentication enabled, for example, user **admin**. They are not required for accessing clusters with Kerberos authentication disabled.

When accessing Manager for the first time, you must add the address to the trusted site list.

Step 13 Prepare the website access address.

1. Obtain the website address format and the role instance according to **Web UIs**.
2. Click **Services**.
3. Click the specified service name, for example, HDFS.
4. Click **Instance** and view **Service IP Address** of **NameNode(Active)**.

Step 14 In the address bar of the browser, enter the website address to access it.

Step 15 When logging out of the website, terminate and close the SSH tunnel.

----End

9 Accessing Manager

9.1 Accessing FusionInsight Manager (MRS 3.*x* or Later)

Scenario

In MRS 3.*x* or later, FusionInsight Manager is used to monitor, configure, and manage clusters. After the cluster is installed, you can use the account to log in to FusionInsight Manager.

NOTE

If you cannot log in to the WebUI of the component, access FusionInsight Manager by referring to **Accessing FusionInsight Manager from an ECS**.

Accessing FusionInsight Manager Using EIP

Step 1 Log in to the MRS management console.

Step 2 In the navigation pane, choose **Clusters** > **Active Clusters**. Click the target cluster name to access the cluster details page.

Step 3 Click **Manager** next to **MRS Manager**. In the displayed dialog box, configure the EIP information.

1. If no EIP is bound during MRS cluster creation, select an available EIP from the drop-down list on the right of **IEP**. If you have bound an EIP when creating a cluster, go to **Step 3.2**.

   NOTE

   If no EIP is available, click **Manage EIP** to one. Then, select the d EIP from the drop-down list on the right of **EIP**.

2. Select the security group to which the security group rule to be added belongs. The security group is configured when the cluster is created.
3. Add a security group rule. By default, the filled-in rule is used to access the EIP. To enable multiple IP address segments to access Manager, see steps **Step**

6 to Step 9. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

4. Select the information to be confirmed and click **OK**.

Step 4 Click **OK**. The Manager login page is displayed.

Step 5 Enter the default username **admin** and the password set during cluster creation, and click **Log In**. The Manager page is displayed.

Step 6 On the MRS management console, choose **Clusters** > **Active Clusters**. Click the target cluster name to access the cluster details page.

NOTE

To grant other users the permission to access Manager, perform **Step 6** to **Step 9** to add the users' public IP addresses to the trusted IP address range.

Step 7 Click **Add Security Group Rule** on the right of **EIP**.

Step 8 On the **Add Security Group Rule** page, add the IP address segment for users to access the public network and select **I confirm that *public network IP*|*port* is a trusted public IP address. I understand that using 0.0.0.0/0. poses security risks**.

By default, the IP address used for accessing the public network is filled. You can change the IP address segment as required. To enable multiple IP address segments, repeat steps **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

Step 9 Click **OK**.

----End

Accessing FusionInsight Manager from an ECS

Step 1 On the MRS management console, click **Clusters**.

Step 2 On the **Active Clusters** page, click the name of the specified cluster.

Record the **AZ**, **VPC**, **MRS ManagerSecurity Group** of the cluster.

Step 3 On the homepage of the management console, choose **Service List** > **Elastic Cloud Server** to switch to the ECS management console and create an ECS.

- The **AZ**, **VPC**, and **Security Group** of the ECS must be the same as those of the cluster to be accessed.
- Select a Windows public image. For example, a standard image **Windows Server 2012 R2 Standard 64bit(40GB)**.
- For details about other configuration parameters, see **Elastic Cloud Server > User Guide > Getting Started > Creating and Logging In to a Windows ECS**.

NOTE

If the security group of the ECS is different from **Default Security Group** of the Master node, you can modify the configuration using either of the following methods:

- Change the security group of the ECS to the default security group of the Master node. For details, see **Elastic Cloud Server** > **User Guide** > **Security Group** > **Changing a Security Group**.
- Add two security group rules to the security groups of the Master and Core nodes to enable the ECS to access the cluster. Set **Protocol** to **TCP**, **Ports** of the two security group rules to **28443** and **20009**, respectively. For details, see **Virtual Private Cloud > User Guide > Security > Security Group > Adding a Security Group Rule**.

Step 4 On the VPC management console, apply for an EIP and bind it to the ECS.

For details, see **Virtual Private Cloud** > **User Guide** > **Elastic IP** > **Assigning an EIP and Binding It to an ECS**.

Step 5 Log in to the ECS.

The Windows system account, password, EIP, and the security group rules are required for logging in to the ECS. For details, see **Elastic Cloud Server > User Guide > Instances > Logging In to a Windows ECS**.

Step 6 On the Windows remote desktop, use your browser to access Manager.

For example, you can use Internet Explorer 11 in the Windows 2012 OS.

The address for accessing Manager is the address of the **MRS Manager** page. Enter the name and password of the cluster user, for example, user **admin**.

NOTE

- If you access Manager with other cluster usernames, change the password upon your first access. The new password must meet the requirements of the current password complexity policies. For details, contact the administrator.
- By default, a user is locked after inputting an incorrect password five consecutive times. The user is automatically unlocked after 5 minutes.

Step 7 Log out of FusionInsight Manager. To log out of Manager, move the cursor to  in the upper right corner and click **Log Out**.

----End

9.2 Accessing MRS Manager MRS 2.1.0 or Earlier)

Scenario

MRS uses FusionInsight Manager to monitor, configure, and manage clusters. You can access FusionInsight Manager by clicking **Access Manager** on the **Dashboard** tab page of your MRS cluster and entering username **admin** and the password configured during cluster creation on the login page that is displayed.

Accessing FusionInsight Manager Using an EIP

Step 1 Log in to the MRS management console.

Step 2 In the navigation pane, choose **Clusters** > **Active Clusters**. Click the target cluster name to access the cluster details page.

Step 3 Click **Access Manager** next to **MRS Manager**. In the displayed dialog box, set **Access Mode** to **EIP**. For details about **Direct Connect**, see **Access Through Direct Connect**.

1. If no EIP is bound during MRS cluster creation, select an available EIP from the drop-down list on the right of **IEP**. If you have bound an EIP when creating a cluster, go to **Step 3.2**.

   NOTE

   If no EIP is available, click **Manage EIP** to one. Then, select the d EIP from the drop-down list on the right of **EIP**.

2. Select the security group to which the security group rule to be added belongs. The security group is configured when the cluster is created.
3. Add a security group rule. By default, your public IP address used for accessing port 9022 is filled in the rule. To enable multiple IP address segments to access MRS Manager, see **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

   NOTE

   - It is normal that the automatically generated public IP address is different from the local IP address and no action is required.
   - If port 9022 is a Knox port, you need to enable the permission of port 9022 to access Knox for accessing MRS Manager.

4. Select the checkbox stating that **I confirm that xx.xx.xx.xx is a trusted public IP address and MRS Manager can be accessed using this IP address**.

Step 4 Click **OK**. The MRS Manager login page is displayed.

Step 5 Enter the default username **admin** and the password set during cluster creation, and click **Log In**. The MRS Manager page is displayed.

Step 6 On the MRS management console, choose **Clusters** > **Active Clusters**, and click the target cluster name to access the cluster details page.

NOTE

To assign MRS Manager access permissions to other users, follow instructions from **Step 6** to **Step 9** to add the users' public IP addresses to the trusted range.

Step 7 Click **Add Security Group Rule** on the right of **EIP**.

Step 8 On the **Add Security Group Rule** page, add the IP address segment for users to access the public network and select **I confirm that the authorized object is a trusted public IP address range. Do not use 0.0.0.0/0. Otherwise, security risks may arise**.

By default, the IP address used for accessing the public network is filled. You can change the IP address segment as required. To enable multiple IP address

segments, repeat steps **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

Step 9 Click **OK**.

----End

Accessing MRS Manager Using an ECS

Step 1 On the MRS management console, click **Clusters**.

Step 2 On the **Active Clusters** page, click the name of the specified cluster.

Record the **AZ**, **VPC**, and **Security Group** of the cluster.

Step 3 On the ECS management console, create an ECS.

- The **AZ**, **VPC**, and **Security Group** of the ECS must be the same as those of the cluster to be accessed.
- Select a Windows public image. For example, select the standard image **Windows Server 2012 R2 Standard 64bit(40GB)**.
- For details about other configuration parameters, see **Elastic Cloud Server > User Guide > Getting Started > Creating and Logging In to a Windows ECS**.

NOTE

If the security group of the ECS is different from **Default Security Group** of the MRS cluster, you can modify the configuration using either of the following methods:

- Change the default security group of the ECS to the security group of the MRS cluster. For details, see **Elastic Cloud Server** > **User Guide** > **Security Group** > **Changing a Security Group**.
- Add two security group rules to the security groups of the Master and Core nodes to enable the ECS to access the cluster. Set **Protocol** to **TCP** and **ports** of the two security group rules to **28443** and **20009**, respectively. For details, see **Virtual Private Cloud** > **User Guide** > **Security** > **Security Group** > **Adding a Security Group Rule**.

Step 4 On the VPC management console, apply for an EIP and bind it to the ECS.

For details, see **Virtual Private Cloud** > **User Guide** > **Elastic IP** > **Assigning an EIP and Binding It to an ECS**.

Step 5 Log in to the ECS.

The Windows system account, password, EIP, and the security group rules are required for logging in to the ECS. For details, see **Elastic Cloud Server > User Guide > Instances > Logging In to a Windows ECS**.

Step 6 On the Windows remote desktop, use your browser to access Manager.

For example, you can use Internet Explorer 11 in the Windows 2012 OS.

The Manager access address is in the format of **https://***Cluster Manager IP Address*:**28443/web**. Enter the name and password of the MRS cluster user, for example, user **admin**.

NOTE

- To obtain the cluster manager IP address, remotely log in to the Master2 node, and run the **ifconfig** command. In the command output, **eth0:wsom** indicates the cluster manager IP address. Record the value of **inet**. If the cluster manager IP address cannot be queried on the Master2 node, switch to the Master1 node to query and record the cluster manager IP address. If there is only one Master node, query and record the cluster manager IP address of the Master node.
- If you access MRS Manager with other MRS cluster usernames, change the password upon your first access. The new password must meet the requirements of the current password complexity policies.
- By default, a user is locked after inputting an incorrect password five consecutive times. The user is automatically unlocked after 5 minutes.

Step 7 Log out of FusionInsight Manager. To log out of Manager, move the cursor to  in the upper right corner and click **Log Out**.

----End

Changing an EIP for a Cluster

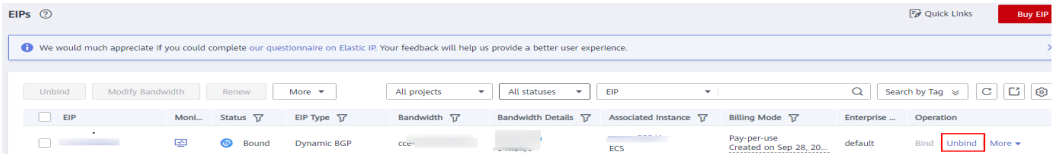
Step 1 On the MRS management console, choose **Clusters** > **Active Clusters**, and click the target cluster name to access the cluster details page.

Step 2 View EIPs

Step 3 Log in to the VPC management console.

Step 4 Choose **Elastic IP and Bandwidth** > **EIPs**.

Step 5 Search for the EIP bound to the MRS cluster and click **Unbind** in the **Operation** column to unbind the EIP from the MRS cluster.



Step 6 Log in to the MRS management console, choose **Clusters** > **Active Clusters**, and click the target cluster name to access the cluster details page.

EIP on the cluster details page is displayed as **Unbound**.

Step 7 Click **Access Manager** next to **MRS Manager**. In the displayed dialog box, set **Access Mode** to **EIP**.

Step 8 Select a new EIP from the EIP drop-down list and configure other parameters. For details, see **Accessing FusionInsight Manager Using an EIP**.

----End

Granting the Permission to Access MRS Manager to Other Users

Step 1 On the MRS management console, choose **Clusters** > **Active Clusters**, and click the target cluster name to access the cluster details page.

Step 2 Click **Add Security Group Rule** on the right of **EIP**.

Step 3 On the **Add Security Group Rule** page, add the IP address segment for users to access the public network and select **I confirm that the authorized object is a trusted public IP address range. Do not use 0.0.0.0/0. Otherwise, security risks may arise**.

By default, the IP address used for accessing the public network is filled. You can change the IP address segment as required. To enable multiple IP address segments, repeat steps **Step 1** to **Step 4**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

Step 4 Click **OK**.

----End

10 FusionInsight Manager Operation Guide (Applicable to 3.x)

10.1 Getting Started

10.1.1 FusionInsight Manager Introduction

Overview

MRS manages and analyzes massive data and helps you rapidly obtain desired data from structured and unstructured data. The structure of open-source components is complex. The installation, configuration, and management processes are time- and labor-consuming. FusionInsight Manager is a unified enterprise-level cluster management platform and provides the following functions:

- **Cluster monitoring**: allows you to better understand status of hosts and services.
- **Graphical indicator monitoring and customization**: allow you to obtain key system information in a timely manner.
- **Service property configuration**: allows you to configure service properties based on the performance requirements of your services.
- **Cluster, service, and role instance operations**: allow you to start or stop services and clusters with just a few clicks.
- **Permission management and audit**: allow you to configure the access control and manage operation logs.

Supported Browsers

- Google Chrome.

  The Google Chrome 93 to 95 versions are recommended.
- Edge

  Supports the Edge browser that comes with the Windows 10 system.

NOTE

It is recommended to access FusionInsight Manager using a browser on the Windows platform.

Introduction to the Manager GUI

FusionInsight Manager provides a unified cluster management platform, facilitating rapid and easy O&M for clusters.

The upper part of the page is the operation bar, the middle part is the display area, and the bottom part is the taskbar.

- **Table 10-1** describes the functions of each portal on the operation bar.

Table 10-1 Functions of each portal on the operation bar

Portal	Function Description
Homepage	Shows the key monitoring indicators and statuses of clusters and the host statuses in column charts, line charts, and tables. You can customize a dashboard for the key monitoring indicators and drag it onto any position on the visualized interface layout. The system overview page supports automatic data update. For details, see **Homepage**.
Cluster	Provides the service monitoring, operation, and configuration guidance, which helps you manage services in a unified manner. For details, see **Cluster**.
Hosts	Provides the host monitoring and operation guidance to help you manage hosts in a unified manner. For details, see **Hosts**.
O&M	Allows you to query and handle alarms, and helps you identify product faults and potential risks in a timely manner, ensuring proper system running. For details, see **O&M**.
Audit	Allows you to query and export audit logs, and view all user activities and operations. For details, see **Audit**.
Tenant Resources	Provides a unified tenant management platform. For details, see **Tenant Resources**.
System	Provides FusionInsight Manager system configuration and management functions, such as user permission configuration. For details, see **System Configuration**.

10.1.2 Querying the FusionInsight Manager Version

Before performing system upgrade and routine maintenance operations, you need to query the current FusionInsight Manager version.

- **Using the GUI**

  On the **Homepage** interface, click  on the upper right corner and choose **About** from the shortcut menu. In the displayed interface, view the version number of FusionInsight Manager.

- **Using the CLI**

  a. Log in to the active management node using the IP address of this node as user **root**.

  b. Run the following command to check the FusionInsight_Manager version and platform information:

  su - omm

  cd ${BIGDATA_HOME}/om-server/om/sbin/pack

  ./queryManager.sh

  Information similar to the following is displayed:

  ```
  Version          Package                          Cputype
  ***              FusionInsight_Manager_***        x86_64
  ```

  NOTE

  The version number *** is subject to the actual version number.

10.1.3 Logging In to FusionInsight Manager

Scenarios

This section describes how to log in to FusionInsight Manager using an account after FusionInsight Manager is installed.

Procedure

Step 1 Obtain the URL of FusionInsight Manager.

Step 2 On login page, enter the username and password.

Step 3 New users need to change their passwords.

The password must meet the following requirements:

- It must contain 8 to 64 characters.
- It must contain at least four of the following character types: uppercase letters, lowercase letters, digits, spaces, and special characters `~!@#$%^&*()-_=+|[{}];',<.>/\?.
- It must be different from the username or its reverse.
- It must be different from the current password.

Step 4 Move your cursor to  in the upper right corner of FusionInsight Manager, and click **Log Out** and and click **OK** from the drop-down list to log out of the current user.

----End

10.1.4 Logging In to the Management Node

Scenarios

Some O&M operation scripts and commands need to be run or can be run only on the active management node. You can identify and log in to the active or standby management node based on the following operations.

Checking and Logging In to the Active and Standby Management Nodes

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **OMS**.

In the **Basic Information** area, **Current Active** indicates the host name of the active management node, and the **Current Standby** indicates the host name of the standby management node.

Click a host name to go to the host details page. On the host details page, record the IP address of the host.

Step 3 Log in to the active or standby management node as user **root**.

----End

Identifying the Active and Standby Management Nodes by Running Scripts and Logging In to Them

Step 1 Log in to any node where FusionInsight Manager is deployed as user **root**.

Step 2 Run the following command to identify the active and standby management nodes:

su - omm

sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh

In the command output, the value of **HAActive** for the active management node is **active**, and that for the standby management node is **standby**. In the following example, **Master1** is the active management node, and **Master2** is the standby management node.

```
HAMode
double
NodeName            HostName        HAVersion          StartTime               HAActive
HAAllResOK          HARunPhase
192-168-0-30        Master1         V100R001C01        xxxx-09-01 07:12:05     active
normal              Actived
192-168-0-24        Master2         V100R001C01        xxxx-09-01 07:14:02     standby
normal              Deactived
```

Step 3 Run the following command to obtain the IP addresses of the active and standby management nodes.

cat /etc/hosts

The following is an example of the IP addresses of the active and standby management nodes:

```
127.0.0.1       localhost
192.168.0.30    Master1
192.168.0.24    Master2
```

Step 4 Log in to the active or standby management node as user **root**.

----End

10.2 Homepage

10.2.1 Overview

After you log in to FusionInsight Manager, the contents on the **Homepage** tab page are displayed by default. The **Summary** page displays the service status preview and the monitoring status report of clusters. The **Alarm Analysis** page displays statistics and analysis on top alarms.

- On the right of the operation bar, you can view the number of alarms of different severities, number of running tasks, current users, and help information.
  - Click  to view the task name, cluster, status, progress, start time, and end time of the latest 100 operation tasks in the **Task Management Center**.

    NOTE

    For a start, stop, restart, or rolling restart task, you can click the task name in the **Task Management Center** and click **Abort**. Then, enter the administrator password as prompted to abort the task. After the task is aborted successfully, the task stops.
  - Click  and choose any help information in the displayed short-cut menu to obtain the details. **Table 10-2** describes the help information in the displayed shortcut menu.

    Table 10-2 List of help information

Item	Description
About	Provides current FusionInsight Manager version number information.

- The taskbar at the bottom of the home page displays the language options of FusionInsight Manager and the current cluster time and time zone information. You can switch the system language.

Service Status Preview Area

On the home page, the number of hosts in clusters and number of installed services are displayed on the left. You can click  to display all service information of a cluster and view status and alarm information of each service installed in the cluster.

Click ⋯ to perform basic O&M operations on the current cluster. For details, see **Table 10-3**.

The icon on the left of each service name indicates that the service is running properly, the icon on the left of each service name indicates that the service is failed to start. The icon indicates that the service is not started.

On the right of the service name, you can check whether an alarm is generated for the service. If an alarm exists, the icon is used to identify the alarm severity and display the number of alarms.

If a component supports multiple services, and multiple services are installed in a cluster, the number of installed services will be displayed on the right of the service.

The icon displayed on the right of the service name indicates that the service configuration has expired.

Monitoring Status Report Area

The chart area is on the right of the **Homepage**, which shows the monitoring reports of the key status, such as the status of all hosts in the cluster, host CPU usage, and host memory usage. You can customize the monitoring reports displayed in the chart area. For details about managing the monitoring indicators, see **Managing the Monitoring Indicator Report**.

The graph data sources are displayed in the lower left. You can zoom in a monitoring report to view the detailed information or close the report.

Alarm Analysis

The **Top 20 Alarms** table and **Analysis on Top 3 Alarms** chart are provided on the **Alarm Analysis** tab page. Click the alarm name in the **Top 20 Alarms** table to display the analysis information of this alarm only. Top alarms and their occurrence time are both provided, so that you could handle alarms accordingly to improve system stability.

10.2.2 Managing the Monitoring Indicator Report

Scenarios

On FusionInsight Manager, you can customize the monitoring items displayed on the **Homepage** page and export monitoring data.

NOTE

The time unit of the horizontal axis varies with the custom duration of historical reports. The details are as follows:

- If the custom duration is 0 to 25 hours, the time unit is 5 minutes. In this case, the cluster must have been installed for at least 10 minutes. The system can reserve monitoring data generated in the latest 15 days.
- If the custom duration is 25 to 150 hours, the time unit is 30 minutes. In this case, the cluster must have been installed for at least 30 minutes. The system can reserve monitoring data generated in the latest 3 months.
- If the custom duration is 150 to 300 hours, the time unit is 1 hour. In this case, the cluster must have been installed for at least 1 hour. The system can reserve monitoring data generated in the latest 3 months.
- If the custom duration is 300 hours to 300 days, the time unit is 1 day. In this case, the cluster must have been installed for at least 1 day. The system can reserve monitoring data generated in the latest 6 months.
- If the custom duration is greater than 300 days, the time unit is 7 days. In this case, the cluster must have been installed for at least 7 days. The system can reserve monitoring data generated in the latest 1 year.
- If the disk usage of the GaussDB partition used by the FusionInsight Manager storage exceeds 80%, the real-time monitoring data and monitoring data whose monitoring period is 5 minutes are cleared.
- For **Storage Resource (HDFS)** in **Tenant Resources**, if the custom duration is 0 to 300 hours, the time unit is 1 hour. In this case, the cluster must have been installed for at least one hour. The system can reserve monitoring data generated in the latest three months.

Customizing the Monitoring Indicator Report

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Homepage**.

Step 3 In the upper right corner of the chart area, click ▼ and choose **Customize** from the displayed menu.

NOTE

The monitoring period is in the unit of five minutes. The monitoring data of the latest one hour is displayed. After the real-time monitoring page is displayed, the real-time monitoring data generated in five minutes is displayed on the right of the monitoring graph.

Step 4 In the navigation tree on the left, select a resource subject to be monitored.

Step 5 Select one or more monitoring indicators from the monitoring list on the right.

Step 6 Click **OK**.

----End

Exporting Monitoring Data

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Homepage**.

Step 3 In the upper right corner of the chart area of the cluster to be operated, select a time range to obtain monitoring data. For example, **1 Week**.

The default setting is real-time monitoring data, which cannot be exported. Click to customize a time range for the monitoring data to be exported.

Step 4 In the upper right corner of the chart area, click and choose **Export** in the displayed menu.

----End

Exporting the Data of Specified Monitoring Items

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Homepage**.

Step 3 In the upper right corner of any monitoring report pane in the chart area of the cluster to be operated, click .

Step 4 Select a time range to obtain monitoring data. For example, **1 Week**.

By default, real-time monitoring data is selected, which cannot be exported. Click to customize a time range for the monitoring data to be exported.

Step 5 Click **Export**.

----End

10.3 Cluster

10.3.1 Cluster Management

10.3.1.1 Overview

Dashboard

Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Dashboard** to view the status of the current cluster.

On the **Dashboard** page, you can start, stop, rolling restart, synchronize configurations for, and perform basic management operations for the current cluster, as described in **Table 10-3**.

Table 10-3 Management and maintenance

UI Portal	Description
Start	Start all services in the cluster.
Stop	Stop all services in the cluster.

UI Portal	Description
More > Restart	Restart all services in the cluster.
More > Rolling-restart Service	Restart all services in the cluster without interrupting services. For details, see **Performing a Rolling Restart of a Cluster**.
More > **Synchronize Configurations**	Synchronize parameter configurations for all services in the cluster.
More > Restart Configuration-Expired Instances	Restart all instances whose configurations have expired. For details, see **Managing Expired Configurations**.
More > **Health Check**	Perform health checks on the OMS, and all services and nodes in the cluster. The health check covers checks on the running status of each object, alarms, and user-defined monitoring metrics. The check result differs from the displayed **Running Status**. To export the result of the health check, click **Export Report** in the upper left corner. If any problem is detected, click **Help**.
More > **Download Client**	Download the default client for the user. For details, see **Downloading the Client**.
More > **Export Installation Template**	Export all installation configurations of the cluster in batches, including the cluster authentication mode, node information, and service configurations. This operation is performed when the cluster needs to be reinstalled in the same environment.
More > **Export Configurations**	Export configurations of all services in the cluster in batches.
More > Enter Maintenance Mode/Exit Maintenance Mode	Configure the cluster to enter or exit the maintenance mode.
More > O&M View	Check the services or hosts that are in the maintenance state.

10.3.1.2 Performing a Rolling Restart of a Cluster

Scenarios

Rolling restart means to restart a cluster without interrupting services after the service role is updated or the configuration is modified in the cluster.

If you need to restart all services in the cluster in batches without interrupting services, you can perform a rolling restart.

NOTE

- Some services do not support a rolling restart. These services will experience a common restart during the rolling restart and may be interrupted. Perform operations as prompted.
- For configurations that must take effect immediately, for example, configuration of the port for a server, a rolling restart is not recommended. Perform a common restart instead.

Impact on the System

Compared with a common restart, a rolling restart does not interrupt services, but it takes longer time than a common restart and may affect throughput and performance of the service to be restarted.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > ***Name of the desired cluster*** **> Dashboard > More > Rolling-restart Service**.

Step 3 In the displayed dialog box, enter the password of the current login user and click **OK**.

Step 4 Set the parameters as required, as shown in **Table 10-4**.

Table 10-4 Rolling restart parameters

Parameter	Description
Restart only instances with expired configurations in the cluster	Specifies whether to restart only the modified instances in a cluster.
Enable rack strategy	Specifies whether to enable the concurrent rolling restart of rack strategy. This option takes effect for roles that meet the rolling restart requirements of the rack strategy. (The roles support the rack-aware function, and instances of the roles belong to two or more racks). **NOTE** This parameter can be set only when a rolling restart is performed on HDFS or YARN.

Parameter	Description
Data Nodes to Be Batch Restarted	Specifies the number of instances that are restarted for each batch when the batch rolling restart strategy is used. The default value is **1**. **NOTE** • This parameter is valid only when the batch rolling restart strategy is used and the instance is the DataNode. • When the rack strategy is enabled, this parameter is invalid. In this case, the cluster uses the default maximum number of instances (20) configured in the rack strategy as the maximum number of instances that are concurrently restarted in a rack. • This parameter can be set only when a rolling restart is performed on HDFS, YARN, Kafka, Storm, or Flume. • This parameter for the RegionServer of HBase cannot be manually configured. Instead, it is automatically adjusted based on the number of RegionServer nodes. Specifically, if the number of RegionServer nodes is less than 30, the parameter value is **1**. If the number is greater than or equal to 30 and less than 300, the parameter value is **2**. If the number is greater than or equal to 300, the parameter value is 1% of the number (rounded-down).
Batch Interval	Specifies the interval between two batches of instances to be rolling restarted. The default value is **0**.
Decommissioning Timeout Interval	Specifies the decommissioning timeout interval for role instances during a rolling restart. The default value is 1800s. Some roles (such as HiveServer and JDBCServer) stop providing services before the rolling restart. Stopped instances cannot establish new connections. Existing connections will be completed after a period of time. A proper configuration of the timeout parameters can minimize the risk of service interruption. **NOTE** This parameter can be set only when a rolling restart is performed for Hive and Spark2x.
Batch Fault Tolerance Threshold	Specifies the tolerance times when the rolling restart of instances fails to be executed in batches. The default value is **0**, which indicates that the rolling restart task ends after any batch of instances fails to be restarted.

NOTE

Set advanced parameters, such as **Data Nodes to Be Batch Restarted**, **Batch Interval**, and **Batch Fault Tolerance Threshold** based on site requirements. Otherwise, services may be interrupted or the performance may be severely affected. Therefore, exercise caution when performing this operation.

The following shows an example:

- If **Data Nodes to Be Batch Restarted** is too large, a great number of instances are restarted at the same time. As a result, services are interrupted or the performance is severely affected because the number of remaining instances is small.
- If **Batch Fault Tolerance Threshold** is too large, services will be interrupted when a new batch of instances is restarted after the previous instance restart failed.

Step 5 Click **OK** and wait until the rolling restart is complete.

----End

10.3.1.3 Managing Expired Configurations

Scenarios

If a new configuration needs to be delivered to all services in the cluster, or **Configuration Status** of multiple services is set to **Expired** or **Failed** after a configuration is modified, the configuration parameters of these services are not synchronized and do not take effect, you can synchronize the configuration and restart related services for the cluster to make new configuration parameters take effect in all services.

If the configuration of the services in the cluster has been synchronized but do not take effect, you need to restart the instances whose configuration has expired.

Impact on the System

- After synchronizing the cluster configuration, you need to restart the services whose configuration has expired. These services are unavailable during restart.
- The instances whose configuration has expired are unavailable during restart.

Procedure

Synchronize configurations.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Dashboard.**

Step 3 Choose **More** > **Synchronize Configurations**.

Step 4 In the displayed dialog box, click **OK**.

----End

Restart instances whose configurations have expired.

Step 1 Choose **More** > **Restart Configuration-Expired Instances**.

Step 2 In the displayed dialog box, enter the password of the current login user and click **OK**.

Step 3 In the displayed dialog box, click **OK**.

You can click **View Instance** to open the list of all expired instances and confirm that the instances have been restarted.

----End

10.3.1.4 Downloading the Client

Scenarios

A default client is provided for MRS clusters. You can manage the cluster, run services, and perform secondary development using this client. Before using the client, you need to download the client software package.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Dashboard** > **More** > **Download Client**.

The **Download Cluster Client** dialog box is displayed.

Step 3 Select a type in the **Select Client Type** dialog box.

- The **Complete Client** type indicates that the package contains scripts, compilation files, and configuration files.
- The **Configuration Files Only** type indicates that the package contains only the client configuration file.

  This type is applicable to application development tasks. For example, it can be used in the following scenario: All client files are downloaded and installed and the administrator modifies the service configuration on FusionInsight Manager. The developer needs to update the client configuration files.

NOTE

There are two platform types: x86_64 and aarch64, which can be installed on the x86 and TaiShan nodes respectively. By default, the platform type of the downloaded client is the same as that of the server.

Step 4 Determine whether to generate a client software package file on the cluster node.

- If yes, select **Save to Path** and click **OK** to generate the client file.

  After the file is generated, it is stored in the **/tmp/FusionInsight-Client/** directory on the primary management node by default. You can also store the client file in other directories, and user **omm** has the read, write, and execute permissions on the directory. If the client file already exists in the path, the existing client file will be replaced.

  After the file is generated, copy the obtained package to another directory, as user **omm** or client installation user, for example, the **/opt/Bigdata/client** directory.
- If no, click **OK** and download the client file to the local PC.

Download the client software package, and wait until the download is complete.

After the client is successfully downloaded, install the client by referring to **Installing a Client**.

----End

10.3.1.5 Modifying Cluster Properties

Scenarios

FusionInsight Manager allows you to view basic attributes after the cluster is installed.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Cluster Properties**.

You can view the cluster name, cluster description, product type, cluster ID, authentication mode, cluster creation time and installed component by default.

Step 3 Change the cluster name.

1. Click and enter a new name.

   The following naming rules are as follows: Enter 2 to 199 characters, including letters, digits, underscores (_), hyphens (-), and spaces. Spaces can be placed only between characters.

2. Click **OK** for the new cluster name to take effect.

Step 4 Modify the cluster description.

1. Click and enter a new description.

   Contains a maximum of 199 characters, including letters, digits, commas (,), periods(.), underscores(_), spaces, or newline characters.

2. Click **OK** for the new description to take effect.

----End

10.3.1.6 Management Cluster Configuration

Scenario

FusionInsight Manager allows you to view the changes of service configuration parameters in a cluster by one click, helping you quickly locate faults and improve configuration management efficiency.

Administrators can quickly view all non-default values of each service in the cluster, non-unified values between instances of the same role, historical records of cluster configuration modification, and parameters whose configuration status is expired in the cluster on the configuration page.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Configurations**.

Step 3 Select an operation page based on the scenario.

- To view all non-default values:
  a. Click **All Non-default Values**. The system displays the parameters that are inconsistent with the default values of each service, role, or instance in the current cluster.

     You can click  next to a parameter value to quickly restore to the default value. You can click  to view the historical modification records of the parameter.

     If a large number of parameters need to be configured, you can filter the parameters in the service filter box in the upper right corner of the page or enter keywords in the search box to search for the parameters.
  b. If you need to change the parameter values of the configuration items, change the values according to the parameter description, and then click **Save**. In the dialog box that is displayed, click **OK**.
- To view all non-unified values:
  a. Click **All Non-uniform Values**. The system displays configuration items with different role, instance group, service, or instance configurations in the current cluster.

     Click  next to the parameter value. In the window that is displayed, you can view the differences.
  b. If you need to change the parameter values, click  to cancel the configuration difference or manually adjust the parameter values, click **OK**, and then click **Save**. In the dialog box that is displayed, click **OK**.
- To check the expiration configurations:
  a. Click **Expired Configurations**. Expired configuration items in the current cluster are displayed.
  b. You can filter services in the upper corner of the page to view the expired configurations of different services, or enter keywords in the search box.
  c. The expired configuration items do not take effect completely. If services are not affected, restart the services or instances whose configuration items have expired.
- To view historical configuration records:
  a. Click **Historical Configurations**. The historical configuration change records of the current cluster are displayed. You can view the parameter value change details, including the service to which the parameter belongs, parameter values before and after the modification, and parameter files.
  b. To restore a configuration change, click **Restore Configuration** in the **Operation** column. In the dialog box that is displayed, click **OK**.

NOTE

Some configuration items take effect only after the corresponding services are restarted. After the configuration is saved, restart the service or instance whose configuration has expired in a timely manner.

----End

10.3.1.7 Static Service Pool

10.3.1.7.1 Static Service Resources

Overview

The resources allocated to each service in a cluster are static service resources. Such services include Flume, HBase, HDFS and Yarn. The total volume of computing resources allocated to each service is fixed, and they are static. A tenant can exclusively use or share a service to obtain the resources required for running this service.

Static Service Pool

Static service pools are used to specify service resource configurations.

Static service pools centrally manage resources applicable to each service.

- They restrict the total volume of resources used for services and dynamically configure the total CPU, I/O, and memory resources applicable to nodes running Flume, HBase, HDFS and Yarn.
- The resources of services are isolated. The services in a cluster are isolated from each other, and the workload of one service has limited impact on other services.

Scheduling Mechanism

The time-based dynamic resource scheduling mechanism enables different volumes of static resources to be configured for services at different time, optimizing service running environments and improving the cluster efficiency.

In a complex cluster environment, multiple services share resources in the cluster, but the resource service period of each service may be different.

The following use a bank customer as an example:

- The HBase query service is heavy in the daytime.
- The query service is light, but the Hive analysis service is heavy at night.

If fixed resources are allocated to each service, the following problems may occur:

- The query service cannot obtain sufficient resources while the resources for the analysis service are idle in the daytime.
- The analysis service cannot obtain sufficient resources while the resources for the query service are idle at night.

As a result, the cluster resource utilization is low and the service capability is weak. Resolve the problem in the following ways:

- Sufficient resources need to be configured for HBase in the daytime.
- Sufficient resources need to be configured for Hive at night.

The time-based dynamic scheduling mechanism can efficiently utilize resources and run tasks.

10.3.1.7.2 Configuring Cluster Static Resources

Scenarios

You can adjust resource base on FusionInsight Manager and customize resource configuration groups if you need to control service resources used on each node in a cluster or the available CPU or I/O quotas on each node at different time segments.

Impact on the System

- After a static service pool is configured, the configuration status of affected services is displayed as **Expired**. You need to restart the services. Services are unavailable during restart.
- After a static service pool is configured, the maximum number of resources used by each service and role instance cannot exceed the upper limit.

Procedure

Modify the Resource Adjustment Base

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Static Service Pool Configurations**.

Step 2 Click **Configurations** in the upper right corner. The page for configuring resource pools is displayed.

Step 3 Change the values of **CPU (%)** and **Memory (%)** in the **System Resource Adjustment Base** area.

Modifying the system resource adjustment base changes the maximum physical CPU and memory usage on nodes by services. If multiple services are deployed on the same node, the maximum physical resource usage of all services cannot exceed the adjusted CPU or memory usage.

Step 4 Click **Next**.

To modify parameters again, click **Previous**.

Modify the Default Resource Configuration Group

Step 5 Click **default**. In the **Configure weight** table, set **CPU LIMIT(%)**, **CPU SHARE(%)**, **I/O(%)**, and **Memory(%)** for each service.

NOTE

- The sum of **CPU LIMIT(%)** and **CPU SHARE(%)** used by all services can exceed 100%.
- The sum of **I/O(%)** used by all services can exceed 100% but not 0.
- The sum of **Memory(%)** used by all services can be greater than, smaller than, or equal to 100%.
- **Memory(%)** cannot take effect dynamically and can only be modified in the default configuration group.
- **CPU LIMIT(%)** is used to configure the ratio of the number of CPU cores that can be used by a service to those can be allocated to related nodes.
- **CPU SHARE(%**) is used to configure the ratio of the time when a service uses a CPU core to the time when other services use the CPU core. That is, the ratio of time when multiple services compete for the same CPU core.

Step 6 Click **Generate detailed configurations based on weight configurations**. FusionInsight Manager generates the actual values of the parameters in the default weight configuration table based on the cluster hardware resources and allocation information.

Step 7 Click **OK**.

In the displayed dialog box, click **OK**.

Add a Customized Resource Configuration Group

Step 8 Determine whether to automatically adjust resource configurations at different time segments.

- If yes, go to **Step 9**.
- If no, use the default configurations, and no further action is required.

Step 9 Click **Configuration**, change the system resource adjustment base values, and click **Next**.

Step 10 Click **Add** to add a resource configuration group.

Step 11 In **Step 1: Scheduling Time**, click **Configuration**.

The page for configuring the time policy is displayed.

Modify the following parameters based on service requirements and click **OK**.

- **Repeat**: If this parameter is selected, the customized resource configuration is applied repeatedly based on the scheduling period. If this parameter is not selected, set the date and time when the configuration of the group of resources can be applied.
- **Repeat Policy**: The available values are **Daily**, **Weekly**, and **Monthly**. This parameter is valid only when **Repeat** is selected.
- **Between**: indicates the time period between the start time and end time when the resource configuration is applied. Set a unique time range. If the time range overlaps with that of an existing group of resource configuration, the time range cannot be saved.

NOTE

- The default group of resource configuration takes effect in all undefined time segments.
- The newly added resource group is a parameter set that takes effect dynamically in a specified time range.
- The newly added resource group can be deleted. A maximum of four resource configuration groups that take effect dynamically can be added.
- Select a repetition policy. If the end time is earlier than the start time, the resource configuration ends in the next day by default. For example, if a validity period ranges from 22: 00 to 06: 00, the customized resource configuration takes effect from 22: 00 on the current day to 06: 00 on the next day.
- If the repeat policy types of multiple configuration groups are different, the time ranges can overlap. The policy types are listed as follows by priority from low to high: daily, weekly, and monthly. The following is an example. There are two resource configuration groups using the monthly and daily policies, respectively. Their application time ranges in a day overlap as follows: 04: 00 to 07: 00 and 06: 00 to 08: 00. In this case, the configuration of the group that uses the monthly policy prevails.
- If the repeat policy types of multiple resource configuration groups are the same, the time ranges of different dates can overlap. For example, if there are two weekly scheduling groups, you can set the same time range on different day for them, such as to 04: 00 to 07: 00, on Monday and Wednesday, respectively.

Step 12 Modify the resource configuration of each service in **Step 2: Weight Configuration**.

Step 13 Click **Generate detailed configurations based on weight configurations**. FusionInsight Manager generates the actual values of the parameters in the default weight configuration table based on the cluster hardware resources and allocation information.

Step 14 Click **OK**.

In the displayed dialog box, click **OK**.

----End

10.3.1.7.3 Viewing Cluster Static Resources

Scenarios

The big data management platform can manage and isolate service resources that are not running on YARN using static service resource pools. The system supports time-based automatic adjustment of static service resource pools. This enables the cluster to automatically adjust the parameter values at different periods to ensure more efficient resource utilization.

System administrators can view the monitoring indicators of resources used by each service in the static service pool on FusionInsight Manager. The monitoring indicators are as follows:

- CPU usage of services
- Total disk I/O read rate of services
- Total disk I/O write rate of services
- Total used memory of services

NOTE

After the multi-tenant function is enabled, the CPU, I/O, and memory usage of all HBase instances can be centrally managed.

Procedure

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Static Service Pool Configurations**.

Step 2 In the configuration group list, click a configuration group, for example, **default**.

Step 3 Check the system resource adjustment base values.

- **System Resource Adjustment Base** indicates the maximum volume of resources that can be used by each node in the cluster. If a node has only one service, the service exclusively occupies the available resources on the node. If a node has multiple services, all services share the available resources on the node.
- **CPU** indicates the maximum number of CPUs that can be used by services on a node.
- **Memory** indicates the maximum memory that can be used by services on a node.

Step 4 In the Chart, view the indicator data chart of cluster service resource usage.

NOTE

- You can click **Add Service to Chart** to add static service resource data of specific services (up to 12 services) to the chart.
- For details about how to manage a single chart, see **Managing the Monitoring Indicator Report**.

----End

10.3.1.8 Client Management

10.3.1.8.1 Managing the Client

Scenario

FusionInsight Manager supports unified management of client installation information in a cluster. After a user downloads and installs the client, FusionInsight Manager automatically records information about the installed (registered) client to facilitate query and management. In addition, you can manually add or modify the information about clients that are not automatically registered, for example, clients installed in earlier versions.

Procedure

Viewing client information

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Client Management** to view the information about the installed clients in the cluster. You can view the IP address,

installation path, component list, registration time, and installation user of the node where the client is located. When you download and install the client of the latest version, the client information is automatically registered.

Adding client information

Step 3 If you need to manually add information for an installed client, click **Add** and manually add the IP address, installation path, user, platform information, and registration information of the client as prompted.

Step 4 Configure the client information and click **OK**.

Modifying client information

Step 5 Information of a manually registered client can be manually modified.

On the **Client Management** page, select the desired client and click **Modify**. After information is modified, click **OK**.

Deleting client information

Step 6 On the **Client Management** page, select the desired client and click **Delete**. In the displayed dialog box, click **OK**.

To delete information of multiple clients, select the desired clients and click **Batch Delete**. In the displayed dialog box, click **OK**.

Exporting client information

Step 7 On the **Client Management** page, click **Export All** to export information about all registered clients to the local PC.

----End

NOTE

On the **Client Management** page, only components that have clients are displayed in the component list. Therefore, some components that do not have clients and special components are not displayed.

The following components are not displayed:

LdapServer, KrbServer, DBService, Hue, Mapreduce, and Flume

10.3.1.8.2 Batch Upgrading Clients

Scenario

The client package downloaded from FusionInsight Manager contains the client batch upgrade tool. If multiple clients need to be upgraded after a cluster upgrade or capacity expansion, you can use this tool to batch upgrade the clients. In addition, the tool provides a lightweight function for batch updating the **/etc/hosts** file on the nodes where the clients are located.

Procedure

Preparation Before the Client Upgrade

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **More** > **Download Client** to download the client package to the specified directory on the server.

For details, see section **Downloading the Client**.

Decompress the downloaded client package and find the **batch_upgrade** directory, for example, **/tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_ClientConfig/batch_upgrade**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **Client Management**. On the **Client Management** page that is displayed, and click **Export All** to export the information about the selected clients to the local PC.

Step 4 Decompress the exported client information and upload the **client-info.cfg** file to the **batch_upgrade** directory.

Step 5 Supplement the ciphertext password in the **client-info.cfg** file by referring to **Reference**.

Batch Upgrading Clients

Step 6 Run the **sh client_batch_upgrade.sh -u -f /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_Client.tar -g /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_ClientConfig/batch_upgrade/client-info.cfg** command to perform the upgrade.

NOTICE

Because the password is configured, you are advised to delete the **client-info.cfg** file as soon as possible after the upgrade.

Step 7 After the upgrade is complete, verify the upgrade result by running the **sh client_batch_upgrade.sh -c** command.

Step 8 If the client is faulty after the upgrade, run the **sh client_batch_upgrade.sh -s** command to roll back the upgrade.

NOTE

- The client batch upgrade tool moves the original client to the backup directory, and then uses the client package specified by the **-f** parameter to install the client again. Therefore, if the original client contains customized contents, manually save the customized contents from the backup directory or move the customized contents to the client directory after the upgrade before you run the **-c** command. Client backup path: *{Original client path}*-backup
- The **-u** parameter is the prerequisite for the **-c** and **-s** commands. You can run the **-c** command to submit or the **-s** command to perform a rollback only after the **-u** command is executed to perform an upgrade.
- You can run the **-u** command multiple times to upgrade only the clients that fail to be upgraded.
- The client batch upgrade tool also supports clients of early versions.
- To upgrade a client installed by a non-root user, the operator must have the read/write permission for the directory where the client is located and its parent directory. Otherwise, the upgrade fails.
- The client package of the **-f** parameter must be a full client. The client package of a single component or some components cannot be used as the input.

----End

Reference

Before upgrading clients in batches, you need to manually configure the user password for remotely logging in to the client node.

Run the **vi client-info.cfg** command to add the user password.

For example:

```
clientIp,clientPath,user,password
10.10.10.100,/home/omm/client /home/omm/client2,omm,password
```

The fields in the configuration file are as follows:

- **clientIp**: Indicates the IP address of the node where the client is located.
- **clientPath**: Indicates the client installation path. Multiple paths are separated by spaces. Note that the path cannot end with a slash (/).
- **user**: Indicates the username of the node.
- **password**: Indicates the user password of the node.

  NOTE

  - Enter a password.
  - If the execution fails, you can check **node.log** in **work_space/log**_*XXX* under the execution directory.

10.3.1.8.3 Updating the hosts File in Batches

Scenario

The client package downloaded from FusionInsight Manager contains the client batch upgrade tool which provides the lightweight functions of upgrading the client in batches and updating the **/etc/hosts** files on the nodes where the clients reside in batches.

Prerequisites

For details about how to prepare for the upgrade, see "Preparation Before the Client Upgrade" in **Procedure**.

Updating the hosts File in Batches

Step 1 Check whether the user configured on the host where the **/etc/hosts** file needs to be updated is user **root**.

- If yes, go to **Step 2** .
- If no, change the configured user to user **root**, and go to **Step 2**.

Step 2 Run the **sh client_batch_upgrade.sh -r -f /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_Client.tar** command to update the **/etc/hosts** file on the nodes where the clients are located in batches.

NOTE

- When you batch update the **/etc/hosts** file, the client package name you entered can be either the name of a complete client package or configuration file name only (recommended).
- The configured user of the host where the **/etc/hosts** file needs to be updated must be user **root**. Otherwise, the update fails.

----End

10.3.2 Managing a Service

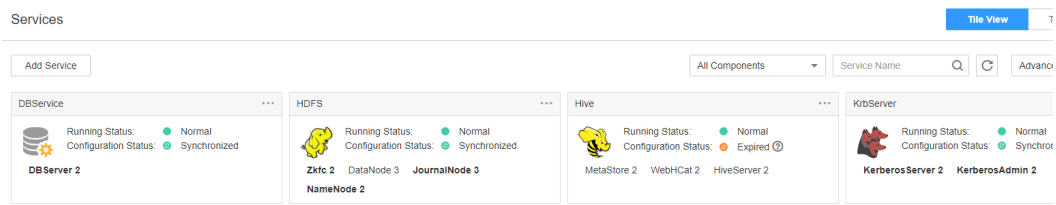
10.3.2.1 Overview

Overview

Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services**.

The service management page containing the function area and service list is displayed.

Figure 10-1 Service management page



Functional Area

In the function area of the service management page, you can select a view type and filter and search by services. You can use the advanced search function to select required services based on the running status.

Service List

The service list on the service management page contains all installed services in the cluster. If the tile view mode is selected, the services will be displayed in pane style. If you select the list view mode, the services will be displayed in a table.

NOTE

In this section, the **Tile View** is used by default.

The service list displays the running status, configuration status, role type, and number of instances of each service. On this page, you can perform some service maintenance tasks, such as starting, stopping, and restarting services.

Table 10-5 Service running status

Status	Description
Normal	Indicates that the service is running properly.
Faulty	Indicates that the service cannot run properly.
Subhealthy	Indicates that some enhanced functions of the service are abnormal.
Not started	Indicates that the service is stopped.
Unknown	Indicates that the initial status of the service cannot be detected.
Starting	Indicates that the service is being started.
Stopping	Indicates that the service is being stopped.
Failed to start	Indicates that the service fails to be started.
Failed to stop	Indicates that the service fails to be stopped.

NOTE

- If the health status of a service is **Faulty**, an alarm is generated. Rectify the fault based on the alarm information.
- HBase, Hive, Spark, and Loader may be in the **Subhealthy** state.
  - If YARN is installed but is abnormal, HBase is in the **Subhealthy** state. If the multi-instance function is enabled, all installed HBase service instances are in the **Subhealthy** state.
  - If HBase is installed but is abnormal, Hive, Spark, and Loader are in the **Subhealthy** state.
  - If any HBase instance is installed but is abnormal after the multi-instance function is enabled, Loader is in the **Subhealthy** state.
  - If an HBase instance is installed but is abnormal after the multi-instance function is enabled, the Hive and Spark instances that map to the HBase instance are in the **Subhealthy** state. That is, if HBase 2 is installed but is abnormal, Hive 2 and Spark2 are in the **Subhealthy** state.

Table 10-6 Service configuration status

Status	Description
Synchronized	Indicates that all service parameter settings have taken effect in the cluster.
Expired	Indicates that the latest configuration is not synchronized and does not take effect after service parameter settings are modified. The configuration needs to be synchronized and the related services need to be restarted. You can click  next to **Configuration Status** to view expired configuration items.
Failed	Indicates that a communication or read/write exception occurs during the parameter configuration synchronization. Use **Synchronize Configuration** to rectify the fault.
Synchronizing	Indicates that the service parameter configuration is being synchronized.
Unknown	Indicates that the initial status of the service cannot be detected.

You can click a service in the service list to perform simple maintenance and management operations on the service, as described in **Table 10-7**.

Table 10-7 Basic management and maintenance

UI Portal	Description
Start Service	Start a specified service in the cluster.
Stop Service	Stop a specified service in the cluster.
Restart Service	Restart a specified service in the cluster. **NOTE** If a service is restarted, other services that depend on this service will be unavailable. Therefore, select **Restart upper-layer services**. Determine whether to perform this operation based on the displayed service list. Services are restarted one by one due to their dependency. **Table 10-8** describes the restart duration of a single service.
Service Rolling Restart	Restart a specified service in the cluster without interrupting services. For details about the parameter settings, see **Table 10-4**.

UI Portal	Description
Synchronize Configuration	• Enable new configuration parameters for a specified service in the cluster. • Deliver new configuration parameters for services whose **Configuration Status** is **Expired**. **NOTE** After some services are synchronized, restart the services for the settings to take effect.

Table 10-8 Restart time

Service Name	Restart Duration	Start Duration	Remarks
CDL	2min	CDLConnector: 1min CDLService: 1min	-
ClickHouse	4min	ClickHouseServer: 2min ClickHouseBalancer: 2min	-
HDFS	10min+x	NameNode: 4 min + x DataNode: 2 min JournalNode: 2 min Zkfc: 2 min	X indicates the duration for loading NameNode metadata. About two minutes are required for each ten million files. For example, if there are 50 million files, x indicates 10 minutes. The start duration is affected by the duration for DataNode to report data blocks.
Yarn	5min+x	ResourceManager: 3 min + x NodeManager: 2 min	x indicates the time corresponding to the number of reserved tasks that need to be restored by ResourceManager. Each 10 thousand reserved tasks require one minute.
MapReduce	2min+x	JobHistoryServer: 2 min + x	x indicates the scanning duration of historical tasks. Each 100 thousand tasks take about 2.5 minutes.
Zookeeper	2min+x	quorumpeer: 2 min + x	x indicates the duration for loading Znodes. Each one million Znodes take about one minute.

Service Name	Restart Duration	Start Duration	Remarks
Hive	3.5min	HiveServer: 3 min MetaStore: 90s WebHcat: 1 min Hiveoverall service: 3 min	-
Spark2x	5min	JobHistory2x: 5 min SparkResource2x: 5 min JDBCServer2x: 5 min	-
Flink	4min	FlinkResource: 1 min FlinkServer: 3min	-
Kafka	2min+x	Broker: 1 min + x	X indicates the data recovery duration. It takes about two minutes to start a single instance with 20000 partitions.
Storm	6min	Nimbus: 3 min UI: 1 min Supervisor: 1 min Logviewer: 1 min	-
Flume	3min	Flume: 2 min MonitorServer: 1 min	-

10.3.2.2 Other Service Management Operations

10.3.2.2.1 Service Details Page

Overview

Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services**. In the service list, click the specified service name to go to the service details page, including the **Dashboard**, **Instance**, **Instance Groups** and **Configurations** tab pages as well as function areas. For some services, the customized management tool page can be displayed.For details about the supported management tools, see**Table 10-9**..

Table 10-9 Customized management tools

Tool	Service	Description
Flume configuration tool	Flume	Configures collection parameters for the Flume server and client.
Flume client management tool	Flume	Views the monitoring information about the Flume client.
Kafka topic monitoring tool	Kafka	Monitors and manages Kafka topics.

The **Dashboard** page is the default page, which contains the basic information, role list, dependency table, and monitoring chart, and more. You can manage services in the upper right corner. For details about basic service management, such as starting, stopping, rolling restart, and synchronization configuration, see **Table 10-7**. For details about other service management operations, see **Table 10-10**.

Table 10-10 Service management operations

Navigation Path	Description
More > **Health Check**	Performs a health check for the current service. The health check items include the health status of each check object, related alarms, and user-defined monitoring indicators. The check result is not the same as the values of **Running Status** displayed on the GUI. To export the result of the health check, click **Export Report** in the upper left corner of the checklist. If you find any problem, click **View Help**.
More > **Download Client**	Download the default client that contains only specific services and perform management operations, run services, or perform secondary development on the client. For details, see **Downloading the Client**.
More > **Change Service Name**	Changes the name of the current service.
More > **Perform** *XX* **Switchover**	For details, see **Performing Active/Standby Switchover of a Role Instance**.
More > **Enter/Exit Maintenance Mode**	Configures a service to enter/exit the maintenance mode.

Navigation Path	Description
Configurations > **Import/Export**	In the scenario where services are migrated to a new cluster or the same services are deployed again, you can import or export all configuration data of a specific service to quickly copy the configuration results.

Basic Information Area

The basic information area on the **Dashboard** tab page contains the basic status data of the service, including the running status, configuration details, version, and key information of the service. If the service supports the open-source web UIs, you can access the open-source web UIs by clicking the links in the basic information area.

NOTE

In the current version, user **admin** does not have the permission to access all the service functions provided on the open source web UI. Create a component service administrator to access the WebUI address.

Role List

The role list on the **Dashboard** tab page contains all roles of the service. The role list displays the running status and the number of instances of each role.

Dependency

The dependency relationship table on the **Dashboard** tab page displays the services on which the current service depends and other services that depend on the service.

Historical Records of Alarms and Events

The alarm and event history area displays the key alarms and events reported by the current service. Up to 20 historical records are displayed.

Chart

The chart area is displayed on the right of the **Dashboard** tab page and contains the key monitoring indicator report of the service. You can customize the monitoring report that is displayed in the chart area, view the description of the monitoring metrics, or export the monitoring data. For a customized resource contribution chart, you can zoom in on the chart and switch between the trend chart and distribution chart.

NOTE

Some services in the cluster provide service-level resource monitoring items. For details, see **Resource Monitoring**.

10.3.2.2.2 Performing Active/Standby Switchover of a Role Instance

Scenarios

Some service roles are deployed in active/standby mode. If the active instance needs to be maintained and cannot provide services, or other maintenance is required, you can manually trigger an active/standby switchover.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 On the service details page, expand the **More** drop-down list and select **Perform** *Role Instance* **Switchover**.

Step 5 In the displayed dialog box, enter the password of the current login user and click **OK**.

Step 6 In the displayed dialog box, click **OK** to perform active/standby switchover for the role instance.

NOTE

- The Manager component package only supports the active/standby switchover of DBService role instances.
- The HD component package supports the active/standby switchover of the following service role instances: HDFS, YARN, Storm, HBase and Mapreduce.
- When an active/standby switchover is performed for a NameNode on HDFS, a NameService must be set.
- The Porter component package only supports the active/standby switchover of Loader role instances.
- This function cannot be used for other role instances.

----End

10.3.2.2.3 Resource Monitoring

Some services in the cluster provide service-level resource monitoring metrics. By default, the monitoring data of the latest 12 hours is displayed. You can click to customize a time range. The default time ranges are as follows: 12 hours, 1 day, 1 week, and 1 month. You can click to export the corresponding report information. If a monitoring item has no data, the report cannot be exported. **Table 10-11** lists the services and monitoring items that support resource monitoring.

Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services**, and click **Resource**. The resource monitoring page is displayed.

Table 10-11 Service resource monitoring

Service	Metrics	Description
HDFS	Resource Usage (by Tenant)	• Collects statistics on HDFS resource usage by tenant. • Views the metrics **Capacity** or **Number of File Objects**.
	Resource Usage (by User)	• Collects statistics on HDFS resource usage by user. • Views the metrics **Used Capacity** or **Number of File Objects**.
	Resource Usage (by Directory)	• Collects statistics on HDFS resource usage by directory. • Views the metrics **Used Capacity** or **Number of File Objects**. • You can click  to configure space monitoring. Alternatively, you can specify an HDFS file system directory for monitoring.
	Resource Usage (by Replica)	• Collects statistics on HDFS resource usages by replica count. • Views the metrics **Used Capacity** or **File Count**.
	Resource Usage (by File Size)	• Collects statistics on HDFS resource usages by file size. • Views the metrics **Used Capacity** or **File Count**.
	Recycle Bin (by User)	• Collects statistics on the usage of the HDFS recycle bin by user. • Views the metrics **Recycle Bin Capacity** or **Number of File Objects**.
	Operation Count	• Collects the number of operations in HDFS.
	Automatic Balancer	• Collects statistics on the execution speed of HDFS automatic balancer and the total capacity of the current balancer migration.
	NameNode RPC Open Connections (by User)	• Displays the number of connections of each user in the Client RPC requests connected to NameNodes.
	Slow DataNodes	Displays DataNode that transmits or processes data slowly in the cluster.

Service	Metrics	Description
	Slow Disks	Displays the disk that processes data slowly on the DataNode in the cluster.
HBase	Operation Requests in Tables	Displays the number of PUT, DELETE, GET, SCAN, INCREMENT, and APPEND operation requests in all tables on all RegionServers.
	Operation Requests on RegionServers	Displays the number of PUT, DELETE, GET, SCAN, INCREMENT, and APPEND operation requests and number of all operation requests in RegionServer.
	Operation Requests for Service	Displays the number of PUT, DELETE, GET, SCAN, INCREMENT, and APPEND operation requests in all regions on RegionServers.
	HFiles on RegionServers	Displays the number of HFiles in all RegionServers.
Hive	HiveServer2-Background-Pool Threads (by IP)	Displays the number of HiveServer2-Background-Pool threads of top users. These threads are measured and displayed in a measurement period.
	HiveServer2-Handler-Pool Threads (by IP)	Displays the number of HiveServer2-Handler-Pools of top users collected and displayed in a period.
	Used MetaStore Number (by IP)	Collects statistics on and displays the MetaStore usage of top users in a period.
	Number of Hive jobs	Displays the number of user-related jobs collected by Hive in a period.
	Number of Files Accessed in the Split Phase	Displays the number of files accessed by the underlying file storage system (HDFS by default) in the Split phase in a period.
	Hive Basic Operation Time	Collects time for creating a directory (mkdirTime), creating a file (touchTime), writing a file (writeFileTime), renaming a file (renameTime), moving a file (moveTime), deleting a file (deleteFileTime), and deleting a directory (deleteCatalogTime) in a period of time.

Service	Metrics	Description
	Table Partitions	Displays the number of partitions in all Hive tables, which is displayed in the following format: *database # table name*, *number of table partitions*.
	HQL Map Count	Collects statistics on HQL statements executed in a period and the number of Map statements invoked during the execution. The displayed information includes users, HQL statements, and the number of Map statements.
	HQL Access Statistics	Displays the number of HQL access times in a period.
Kafka	Kafka Disk Usage Distribution	Displays the disk usage distribution statistics of the Kafka cluster.
Spark2x	HQL Access Statistics	Collects HQL access statistics in a period, including the username, HQL statement, and HQL statement execution times.
Yarn	Used resources (by task)	• Displays the number of CPU cores and memory used by a task. • Views the metrics **By memory** or **By CPU**.
	Resource usage (by tenant)	• Displays the number of CPU cores and memory used by a tenant. • Views the metrics **By memory** or **By CPU**.
	Resource usage ratio (by tenant)	• Displays the ratio of the number of CPU cores to the memory used by a tenant. • Views the metrics **By memory** or **By CPU**.
	Task Duration Ranking	Displays Yarn tasks sorted by time consumption.
	ResourceManager RPC Open Connections (by User)	Displays the number of client RPC connections to ResourceManager by user.
	Operation Count	Collects statistics on the number and proportion of operations corresponding to each Yarn operation type.

Service	Metrics	Description
	Ranking of Tasks in a Queue by Resource Usage	• Displays the resources consumed by the tasks running in a queue after the queue (tenant) is selected on the GUI. • Views the metrics **By memory** or **By CPU**.
	Ranking of Users in a Queue by Resource Usage	• Displays the resources consumed by the users who are running tasks in the queue after a queue (tenant) is selected on the GUI. • Views the metrics **By memory** or **By CPU**.
ZooKeeper	Used Resources (By Second-Level Znode)	• Displays the ZooKeeper level-2 znode resource status. • Views the metrics **By Znode quantity** or **By capacity**.
	Number of Connections (by Client IP Address)	Displays the ZooKeeper client connection resource status.

10.3.2.2.4 Collecting Stack Information

Scenario

To meet the project requirements, the administrator can collect the stack information about a specified role or instance on FusionInsight Manager, save the information to a local directory, and download the information. The following information can be collected:

1. jstack information.
2. jmap -histo information.
3. jmap -dump information.
4. jstack and jmap-histo information can be collected consecutively for comparison.

Procedure

Collecting stack information

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > *Name of the desired service*.

Step 3 Choose **More** > **Collect Stack Information**.

NOTE

- To collect stack information of multiple instances, select the desired instances in the instance list and choose **More** > **Collect Stack Information**.
- To collect stack information of a single instance, click the desired instance and choose **More** > **Collect Stack Information**.

Step 4 In the dialog box that is displayed, select the desired role and content, configure advanced options (retain the default settings unless otherwise specified), and click **OK**.

Step 5 After the collection is successful, click **Download**.

Downloading stack information

Step 6 Choose **Cluster** > *Name of the desired cluster* > **Services** > *Name of the desired service*. Choose **More** > **Download Stack Information** in the upper right corner.

Step 7 Select the desired role and content and click **Download** to download the stack information to the local PC.

Clearing stack information

Step 8 Choose **Cluster** > *Name of the desired cluster* > **Services** > *Name of the desired service*.

Step 9 Choose **More** > **Clear Stack Information** in the upper right corner.

Step 10 Select the desired role and content and configure **File Directory**. Click **OK**.

----End

10.3.2.2.5 Switching Ranger Authentication

Scenarios

By default, the Ranger service is installed and Ranger authentication is enabled for a newly installed cluster in security mode. You can set fine-grained security access policies for accessing component resources through the permission plug-in of the component. If Ranger authentication is not required, the administrator can manually disable Ranger authentication on the service page. After Ranger authentication is disabled, the system continues to perform permission control based on the role model of FusionInsight Manager when accessing component resources.

In a cluster upgraded from an earlier version, Ranger authentication is not used by default when users access component resources. The administrator can manually enable Ranger authentication after installing the Ranger service.

NOTE

- In a cluster in security mode, the following components support Ranger authentication: HDFS, Yarn, Kafka, Hive, HBase, Storm, Spark2x, Impala.
- In a cluster in non-security mode, the Ranger supports permission control on component resources based on OS users. The following components support Ranger authentication: HBase, HDFS, Hive, Spark2x, and Yarn.
- After Ranger authentication is enabled, all authentication of the component will be managed by Ranger. The permissions set by the original authentication plug-in will become invalid (The ACL rules of HDFS and Yarn components still take effect). Exercise caution when performing this operation. You are advised to deploy permissions on Ranger in advance. Please restart the service for the modification to take effect.
- After Ranger authentication is disabled, all authentication of the component will be managed by the permission plug-in of the component. The permission set on Ranger will become invalid. Exercise caution when performing this operation. You are advised to deploy permissions on Manager in advance. Please restart the service for the modification to take effect.

Enabling Ranger Authentication

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 On the service details page, expand the **More** drop-down list and select **Enable Ranger**.

Step 5 In the displayed dialog box, enter the password of the current login user and click **OK**.

Step 6 In the service list, restart the service whose configuration has expired.

----End

Disabling Ranger Authentication

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 On the service details page, expand the **More** drop-down list and select **Disable Ranger**.

Step 5 In the displayed dialog box, enter the password of the current login user and click **OK**. In the displayed dialog box, click **OK**.

Step 6 In the service list, restart the service whose configuration has expired.

----End

10.3.2.3 Service Configuration

10.3.2.3.1 Modifying Service Configuration Parameters

Scenarios

To meet site requirements, you can view and modify default configurations of a service on FusionInsight Manager. Configure parameters based on the information provided in the configuration description.

NOTE

The parameters of DBService cannot be modified when only one DBService role instance exists in the cluster.

Impact on the System

- After configuring properties of a service, you need to restart the service. The service is unavailable during restart. If the instance is not restarted, the configuration status of the instance is **Expired**.
- After the service configuration parameters are modified and then take effect after restart, you need to download and install the client again or download the configuration file to update the client. For example, you can modify configuration parameters of the following services: HBase, HDFS, Hive, Spark, YARN, and MapReduce.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

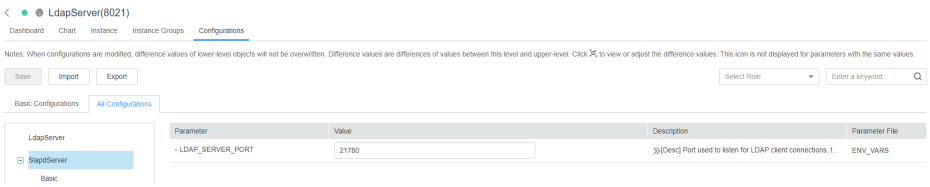
Step 3 Click the specified service name on the service management page.

Step 4 Click **Configurations**.

The **Basic Configuration** tab page is displayed by default. To modify more parameters, click the **All Configurations** tab. The navigation tree displays all configuration parameters of the service. The level-1 nodes in the navigation tree are service names or role names. The parameter category is displayed after the level-1 node is expanded.

As shown in the following figure, the first node **LdapServer** indicates the service name, and the second node **SlapdServer** indicates the role name. The configuration parameter displayed takes effect for all instances of the role and the service.

Figure 10-2 Configuration parameter navigation tree



Step 5 In the navigation tree, select the specified parameter category and change the parameter values on the right.

NOTE

Select a port parameter value from the value range on the right. Ensure that all parameter values in the same service are within the value range and are unique. Otherwise, the service fails to be started.

If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The system searches for the parameter in real time and displays the result.

Step 6 Click **Save**. In the confirmation dialog box, click **OK**.

Wait until the message "Operation succeeded." is displayed. Click **Finish**.

The configuration is modified.

NOTE

- To update the queue configuration of the YARN service without restarting service, choose **More** > **Refresh Queue** to update the queue for the configuration to take effect.
- During configuration of the **flume.config.file** parameter, you can upload and download files. After a configuration file is uploaded, the old file will be overwritten. If the configuration is not saved and the service is restarted, the configuration does not take effect. Save the configuration in time.
- If you need to restart the service for the configuration to take effect after modifying service configuration parameters, choose **More** > **Restart Service** in the upper right corner of the service page.

----End

10.3.2.3.2 Modifying Customized Configuration Parameters of a Service

Scenarios

All open source parameters can be configured for all MRS cluster components. Parameters used in some key application scenarios can be modified on FusionInsight Manager, and some parameters of open source features may not be configured for some component clients. To modify the component parameters that are not directly supported by FusionInsight Manager, you can add new parameters for components by using the configuration customization function on FusionInsight Manager. Newly added parameters are saved in component configuration files and take effect after restart.

Impact on the System

- After configuring properties of a service, you need to restart the service. The service is unavailable during restart. If the instance is not restarted, the configuration status of the instance is **Expired**.
- After the service configuration parameters are modified and then take effect after restart, you need to download and install the client again or download the configuration file to update the client.

Prerequisites

You have understood the meanings of parameters to be added, configuration files to take effect, and impact on components.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 Choose **Configurations** > **All Configurations**.

Step 5 In the navigation tree on the left, locate a level-1 node and select **Customization**. The system displays the customized parameters of the current component.

The configuration files that save the newly added customized parameters are displayed in **Parameter File**. Different configuration files may have same open source parameters. After the parameters in different files are set to different values, the configuration takes effect depends on the loading sequence of the configuration files by components. You can customize parameters for services and roles as required. Adding customized parameters for a single role instance is not supported.

Step 6 Locate the row where a specified parameter resides, enter the parameter name supported by the component in the **Name** column and enter the parameter value in the **Value** column.

You can click **+** or **-** to add or delete a customized parameter.

Step 7 Click **Save**. In the displayed **Save Configuration** dialog box, confirm the modification and click **OK**. Wait until the message "Operation succeeded" is displayed, and click **Finish**.

The configuration is saved successfully.

After the configuration is saved, restart the expired service or instance for the configuration to take effect.

----End

Task Example (Configuring Customized Hive Parameters)

Hive depends on HDFS. By default, Hive accesses the HDFS client. The configuration parameters that have taken effect are controlled by HDFS. For example, the HDFS parameter **ipc.client.rpc.timeout** affects the RPC timeout interval for all the clients (including service and cluster clients) to connect to the HDFS server. To change the timeout interval for Hive to connect to the HDFS, you can change the timeout interval by configuring required customized parameters. After this parameter is added to the **core-site.xml** file of Hive, this parameter can be identified by the Hive service and its configuration overwrites the parameter configuration in HDFS.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 2 Choose **Hive** > **Configurations**. On the displayed page, click the **All Configurations** tab.

Step 3 In the navigation tree on the left, select **Customization** for the Hive service. The system displays the customized service parameters supported by Hive.

Step 4 In **core-site.xml**, locate the row that contains the **core.site.customized.configs** parameter, enter **ipc.client.rpc.timeout** in the **Name** column, and enter a new value in the **Value** column, for example, 150000. The unit is ms.

Step 5 Click **Save**. In the displayed **Save Configuration** dialog box, confirm the modification and click **OK**. Wait until the message "Operation succeeded" is displayed, and click **Finish**.

The configuration is saved successfully.

After the configuration is saved, restart the expired service or instance for the configuration to take effect.

----End

10.3.3 Instance Management

10.3.3.1 Instance Management Overview

Overview

Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **KrbServer** > **Instance**. The displayed instance management page contains the function area and role instance list.

Functional Area

The function area allows you to perform maintenance operations on roles, such as adding roles and starting or stopping instances.

Table 10-12 Instance management and maintenance

UI Portal	Description
Start Instance	Start a specified instance in the cluster. You can start a role instance in the **Not Started**, **Stop Failed**, or **Startup Failed** state to use the role instance.
More > **Stop Instance**	Stop a specified instance in the cluster. Stop role instances that will no longer be used or are abnormal.
More > **Restart Instance**	Restart a specified instance in the cluster. Restart an abnormal role instance to recover its functions.
More > **Instance Rolling Restart**	Restart a specified instance in the cluster without interrupting services. For details about the parameter settings, see **Performing a Rolling Restart of a Cluster**.

UI Portal	Description
More > Decommission/ Recommission	Recommission or decommission a specified instance in the cluster to change the service availability status of the service. For details, see **Decommissioning and Recommissioning an Instance**. **NOTE** Only the role DataNode in HDFS, the role NodeManager in Yarn, the role RegionServe in HBase support the recommissioning and decommissioning functions.
Desired instance > **More** > **Synchronize Configuration**	If **Configuration Status** of a role instance is **Expired**, the role instance is not restarted after its configurations are modified. The new configuration is saved only on FusionInsight Manager. In this case, use this function to deliver the new configuration to the specified instance. **NOTE** • After synchronizing a role instance configuration, restart the role instance whose configuration has expired. The role instance is unavailable during restart. • After the synchronization is complete, restart the instance for the configuration to take effect.
Desired instance > **Instance Configurations**	For details, see **Managing Instance Configurations**.

You can filter instances based on the role they belong to or their running status in this area.

NOTE

Click **Advanced Search** to search for specific instances by specifying filter criteria, such as **Host Name**, **Management IP Address**, **Business IP Address**, or **Instance Groups**.

Role Instance List

The role instance list contains the instances of all roles in the cluster. The list displays the running status, configuration status, hosts, and related IP addresses of each instance.

Table 10-13 Instance running status

Status	Description
Normal	Indicates that the instance is running properly.
Faulty	Indicates that the instance cannot run properly.
Decommissioned	Indicates that the instance is out of service.

Status	Description
Not started	Indicates that the instance is stopped.
Unknown	Indicates that the initial status of the instance cannot be detected.
Starting	Indicates that the instance is being started.
Stopping	Indicates that the instance is being stopped.
Restoring	Indicates that an exception may occur in the instance and the instance is being automatically rectified.
Decommissioning	Indicates that the instance is being decommissioned.
Recommissioning	Indicates that the instance is being recommissioned.
Failed to start	Indicates that the service fails to be started.
Failed to stop	Indicates that the service fails to be stopped.

You can click an instance name to go to the instance details page and view basic information about the instance and the monitoring indicator report of the instance.

10.3.3.2 Decommissioning and Recommissioning an Instance

Scenario

Some role instances provide services for external services in distributed and parallel mode. Services independently store information about whether each instance can be used. Therefore, you need to use FusionInsight Manager to recommission or decommission these instances to change the instance running status.

Some instances do not support the recommissioning and decommissioning functions.

NOTE

The following roles support decommissioning and recommissioning: HDFS DataNode, Yarn NodeManager, and HBase RegionServer.

- If the number of the DataNodes is less than or equal to that of HDFS replicas, decommissioning cannot be performed. If the number of HDFS replicas is three and the number of DataNodes is less than four in the system, decommissioning cannot be performed. In this case, an error will be reported and force FusionInsight Manager to exit the decommissioning 30 minutes after FusionInsight Manager attempts to perform the decommissioning.
- During MapReduce task execution, files with 10 replicas are generated. Therefore, if the number of DataNode instances is less than 10, decommissioning cannot be performed.
- If the number of DataNode racks (the number of racks is determined by the number of racks configured for each DataNode) is greater than 1 before the decommissioning, and after some DataNodes are decommissioned, that of the remaining DataNodes changes to 1, the decommissioning will fail. Therefore, before decommissioning DataNode instances, you need to evaluate the impact of decommissioning on the number of racks to adjust the DataNodes to be decommissioned.
- If multiple DataNodes are decommissioned at the same time, and each of them stores a large volume of data, the DataNodes may fail to be decommissioned due to timeout. To avoid this problem, it is recommended that one DataNode be decommissioned each time and multiple decommissioning operations be performed.

Procedure

Step 1 Perform the following steps to perform a health check for the DataNodes before decommissioning:

1. Log in to the client installation node as a client user and switch to the client installation directory.
2. For a security cluster, use user **hdfs** for permission authentication.

   ```
   source bigdata_env          #Configure client environment variables.
   kinit hdfs                  #Configure kinit authentication.
   Password for hdfs@HADOOP.COM:    #Enter the login password of user hdfs.
   ```

3. Run the **hdfs fsck / -list-corruptfileblocks** command, and check the returned result.
   - If "has 0 CORRUPT files" is displayed, go to **Step 2**.
   - If the result does not contain "has 0 CORRUPT files" and the name of the damaged file is returned, go to **Step 1.4**.
4. Run the **hdfs dfs -rm** *Name of the damaged file* command to delete the damaged file.

   NOTE

   Deleting a file or folder is a high-risk operation. Ensure that the file or folder is no longer required before performing this operation.

Step 2 Log in to FusionInsight Manager.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 4 Click the specified service name on the service management page. On the displayed page, click the **Instance** tab.

Step 5 Select the specified role instance to be decommissioned.

Step 6 Select **Decommission** or **Recommission** from the **More** drop-down list.

In the displayed dialog box, enter the password of the current login user and click **OK**.

Select **I confirm to decommission these instances and accept the consequence of service performance deterioration** and click **OK** to perform the corresponding operation.

NOTE

During the instance decommissioning, if the service corresponding to the instance is restarted in the cluster using another browser, FusionInsight Manager displays a message indicating that the instance decommissioning is stopped, but the operating status of the instance is displayed as **Started**. In this case, the instance has been decommissioned on the background. You need to decommission the instance again to synchronize the operating status.

----End

10.3.3.3 Managing Instance Configurations

Scenarios

You can modify configuration parameters for each role instance. In the scenario where instances are migrated to a new cluster or the corresponding service needs to be deployed again, you can import or export all configuration data of a service on FusionInsight Manager to quickly copy configuration results.

FusionInsight Manager can manage configuration parameters of a single role instance. Modifying configuration parameters and importing or exporting instance configurations do not affect other instances.

Impact on the System

After modifying the configuration of a role instance, you need to restart the instance. The role instance is unavailable during restart. If the instance is not restarted, the configuration status of the instance is **Expired**.

Modifying Instance Configuration

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page. On the displayed page, click the **Instance** tab.

Step 4 Click the specified instance and select **Instance Configurations**.

By default, **Basic Configuration** is displayed. To modify more parameters, select **All Configurations**. All parameter categories supported by the instance are displayed on the **All Configurations** tab page.

Step 5 In the navigation tree, select the specified parameter category and change the parameter values on the right.

If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The system searches for the parameter in real time and displays the result.

Step 6 Click **Save**. In the confirmation dialog box, click **OK**.

Wait until the message "Operation succeeded." is displayed. Click **Finish**.

The configuration is modified.

----End

Exporting/Importing Instance Configuration

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page. On the displayed page, click the **Instance** tab.

Step 4 Click the specified instance and select **Instance Configurations**.

Step 5 Click **Export** to export the configuration parameter file to the local host.

Step 6 On the **Instance Configurations** page, click **Import**, select the configuration parameter file of the instance, and import the file.

----End

10.3.3.4 Viewing the Instance Configuration File

Scenario

FusionInsight Manager allows O&M personnel to view the content configuration files such as environment variables and role configurations of the instance node on the management page. If O&M personnel need to quickly check whether configuration items of the instance are incorrectly configured or when some hidden configuration items need to be viewed, the O&M personnel can directly view the configuration files on FusionInsight Manager. In this case, users quickly analyze configuration problems.

Procedure

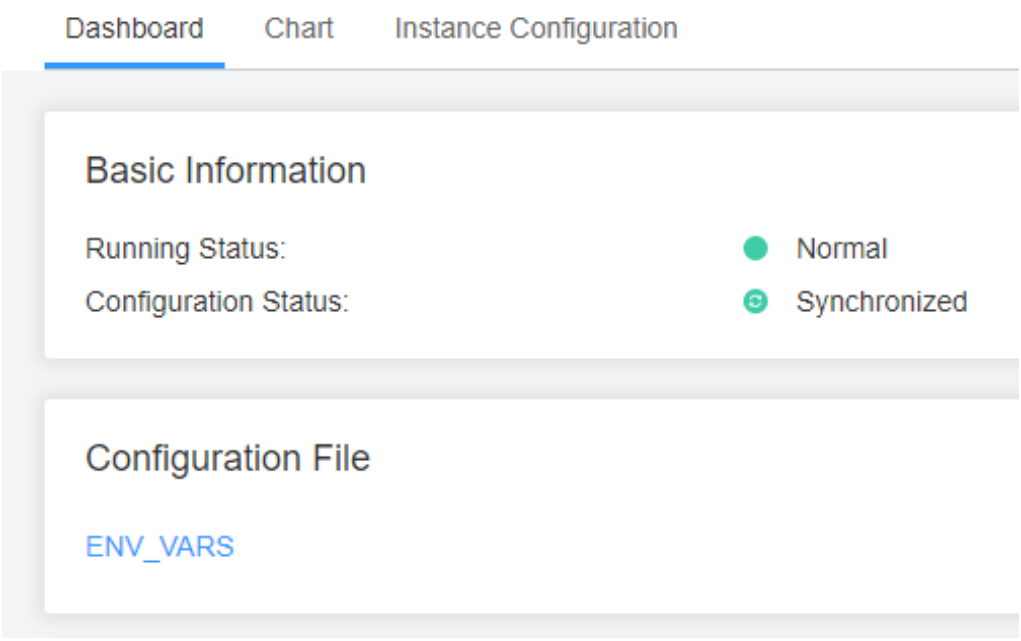
Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page. On the displayed page, click the **Instance** tab.

Step 4 Click the name of the target instance. In the **Configuration File** area on the **Instance Status** page, the configuration file list of the instance is displayed.

Figure 10-3 Viewing the instance configuration file



Step 5 Click the name of the configuration file to be viewed to view the parameter values in the configuration file.

To obtain the configuration file, you can download the configuration file to the local PC.

NOTE

If a node in the cluster is faulty, the configuration file cannot be viewed. Rectify the fault before viewing the configuration file again.

----End

10.3.3.5 Instance Group

10.3.3.5.1 Managing Instance Groups

Scenarios

Instance groups can be managed on FusionInsight Manager. That is, you can group multiple instances in the same role based on a specified principle, such as the nodes with the same hardware configuration. The modification on the configuration parameters of an instance group applies to all instances in the group.

In a large cluster, instance groups are used to improve the capability of managing instances in batches in the heterogeneous environment. After instances are grouped, the instances can be configured repeatedly to reduce redundant instance configuration items and improve system performance.

Creating an Instance Group

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 On the displayed page, click the **Instance Groups** tab.

Click  and configure parameters as prompted.

Table 10-14 Instance group configuration parameters

Parameter	Description
Group Name	Indicates the instance group name. The value can contain only letters, digits, underscores (_), hyphens (-), and spaces. It must start with a letter, digit, underscore (_), or hyphen (-) and cannot ends with a space. It can contain a maximum of 99 characters.
Role	Indicates the role to which an instance group belongs.
Copy From	Indicates that the parameter values of a specified instance group are copied to the parameters of a new group. If the value is null, the default values are used for the parameters of the new group.
Description	Indicates the instance group description. It can contain only letters, digits, commas (,), periods (.), underscores (_), spaces, and line breaks, and can contain a maximum of 200 characters.

NOTE

- Each instance must belong to only one instance group. When an instance is installed for the first time, it belongs to the instance group *Role name*-**DEFAULT** by default.
- You can delete unnecessary or unused instance groups. Before deleting an instance group, migrate all instances in the group to other instance groups, see **Deleting an Instance Group** to delete instance group. The default instance group cannot be deleted.

Step 5 Click **OK**.

The instance group is created.

----End

Modifying Properties of an Instance Group

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 Click the **Instance Groups** tab. On the **Instance Groups** tab page, locate the row that contains the target instance group.

Click  and configure parameters as prompted.

Step 5 Click **OK** to save the modifications.

The default instance group cannot be modified.

----End

Deleting an Instance Group

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 Click the **Instance Groups** tab. On the **Instance Groups** tab page, locate the row that contains the target instance group.

Step 5 Click  .

Step 6 In the displayed dialog box, click **OK**.

The default instance group cannot be deleted.

----End

10.3.3.5.2 Viewing Information About an Instance Group

Scenarios

You can view the instance group of a specified service on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 On the displayed page, click the **Instance Groups** tab.

Step 5 In the navigation tree, select a role. On the **Basic** tab page, view all instances in the instance group.

NOTE

To move an instance from an instance group to another, perform the following operations:

1. Select the instance to be moved and click **Move**.
2. In the displayed dialog box, select an instance group to which the instance to be moved.

   During the migration, the configuration of the new instance group is automatically inherited. If the instance configuration is modified before the migration, the configuration of the instance prevails.
3. Click **OK**.

   Restart the expired service or instance for the configuration to take effect.

----End

10.3.3.5.3 Configuring Instantiation Group Parameters

Scenarios

In a large cluster, users can configure parameters for multiple instances in batches by configuring the related instance groups on FusionInsight Manager, reducing redundant instance configuration items and improving system performance.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 3 Click the specified service name on the service management page.

Step 4 On the displayed page, click the **Instance Groups** tab.

Step 5 In the navigation tree, select the instance group name of a role, and switch to the **Configuration** tab page. Adjust parameters to be modified, and click **Save**. The configuration takes effect for all instances in the instance group.

----End

10.4 Hosts

10.4.1 Host Management Page

10.4.1.1 Viewing the Host List

Overview

Log in to FusionInsight Manager, and click **Hosts**. The host list is displayed on the host management page, you can view the host list and basic information of each host.

You can switch view types and set search criteria to filter and search for hosts.

Host View

Click **Role View** to view the roles deployed on each host. If the role supports the active/standby mode, the role name is displayed in bold.

Host List

The host list on the host management page contains all hosts in the clusters, and O&M operations can be performed on these hosts.

On the host management page, you can filter hosts by node type or cluster. The rules for filtering host types are as follows:

- A Management Node is the node where the OMS is deployed. Additionally, control roles and data roles may also be deployed on Management Nodes.
- A Control Node is the node where control roles are deployed. Additionally, data roles may also be deployed on Control Nodes.
- A Data Node is the node where only data roles are deployed.

If you select the **Host View**, the IP address, rack planning, AZ name, running status, cluster name, and hardware resource usage of each host are displayed.

Table 10-15 Host running status

Status	Description
Normal	Indicates that the host is in the normal state.
Faulty	Indicates that the host is abnormal.
Unknown	Indicates that the initial status of the host cannot be detected.
Isolated	Indicates that the host is isolated.
Suspended	Indicates that the host is suspended.

10.4.1.2 Viewing the Host Dashboard

Overview

Log in to FusionInsight Manager, click **Hosts**, and click a host name in the host list. The host details page contains the basic information area, disk status area, role list area, and monitoring chart.

Basic Information Area

The basic information area contains the key information about the host, such as the management IP address, service IP address, host type, rack, firewall, number of CPU cores, and OS information.

Disk Status Area

The disk status area contains all disk partitions configured for the cluster on the host and the usage of each disk partition.

Instance List Area

The instance list area displays all role instances installed on the host and the status of each role instance. You can click the log file next to a role instance name to view the log file content of the instance online.

Alarm and Event History

The alarm and event history area displays the key alarms and events reported by the current host. The system can display a maximum of 20 history records.

Chart

The monitoring chart area is displayed on the right of the host details page, and contains the key monitoring metrics of the host.

You can click > **Customize** in the upper right corner to customize the monitoring report to be displayed in the chart area. Select a time range and click > **Export** to export detailed monitoring indicator data within the specified time range.

You can click next to the title of a monitoring indicator to open the description of the monitoring indicator.

Click the **Chart** tab of the host to view the full monitoring chart information about the host.

GPU Card Status Area

If the host is configured with GPU cards, the GPU card status area displays the model, location, and status of the GPU card installed on the host.

10.4.1.3 Checking Processes and Resources on the Active Node

Overview

Log in to FusionInsight Manager, click **Hosts**, and click the specified host name in the host list. On the host details page, click the **Process** and **Resource** tabs.

Process

On the **Process** tab page, the information about the role processes of the deployed service instances on the current host is displayed, including the process status, PID, and process running time. You can directly view the log files of each process online.

Resource

On the **Resource** tab page, the detailed resource usage of deployed service instances on the current host is displayed, including the CPU, memory, disk, and port usage.

10.4.2 Host Maintenance Operations

10.4.2.1 Starting and Stopping All Instances on a Host

Scenarios

If a host is faulty, you may need to stop all the roles on the host and perform maintenance check on the host. After the host fault is rectified, start all roles running on the host to recover host services. You can start or stop all instances on a host on the host management page or host details page on FusionInsight Manager. The following describes how to perform such operations on the host management page.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Hosts**.

Step 3 Select the check box of the target host.

Step 4 Select **Start All Instances** or **Stop All Instances** from the **More** drop-down list to start or stop all role instances.

----End

10.4.2.2 Performing a Host Health Check

Scenarios

If the running status of a host is not **Normal**, you can perform health checks on the host to check whether some basic functions are abnormal. During routine O&M, you can perform host health checks to ensure that the configuration parameters and monitoring of each role instance on the host are normal and can run stably for a long time.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Hosts**.

Step 3 Select the check box of the target host.

Step 4 Select **Health Check** from the **More** drop-down list to start the health check.

To export the result of the health check, click **Export Report** in the upper left corner. If any problem is detected, click **Help**.

----End

10.4.2.3 Configuring Racks for Hosts

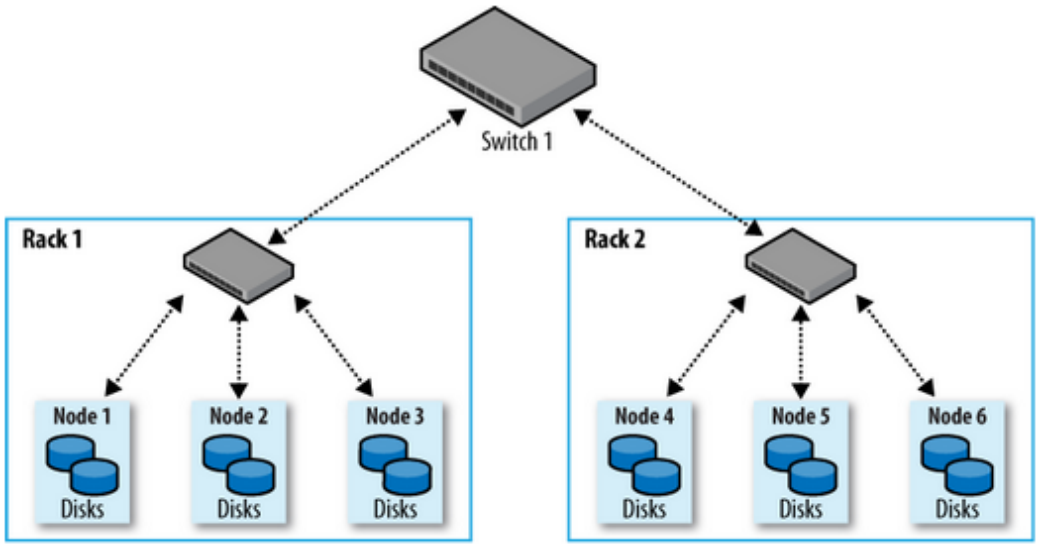
Scenarios

All hosts in a large cluster are usually deployed on multiple racks. Hosts on different racks communicate with each other through switches. The network bandwidth between different hosts on the same rack is much greater than that on different racks. In this case, plan the network topology based on the following requirements:

- To improve the communication speed, it is recommended that data be exchanged between hosts on the same rack.
- To improve the fault tolerance capability, distribute processes or data of distributed services on different hosts of multiple racks as dispersedly as possible.

Hadoop uses a file directory structure to represent hosts. **Figure 10-4** displays a cluster with a two-layer network structure. In this example, you are advised to name the rack for Node 1 as /Switch1/Rack1 and that for Node 4 as /Switch1/Rack2.

Figure 10-4 Two-layer network structure



The HDFS cannot automatically determine the network topology of each DataNode in the cluster. You need to set the rack name to identify the rack where the host is located so that the NameNode can draw the network topology of the required DataNodes and back up data of the DataNodes to different racks. Similarly, YARN needs to obtain rack information and allocate tasks to different NodeManagers as required.

If the cluster network topology changes, you need to reallocate racks for hosts on FusionInsight Manager so that related services can be automatically adjusted.

Impact on the System

If the name of the host rack is changed, policy for storing HDFS replicas, Yarn task assignment, and storage location of Kafka partitions will be affected. After the modification, restart the HDFS, Yarn, and Kafka for the configuration to take effect.

Improper rack configuration will unbalance loads (including CPU, memory, disk, and network) among nodes in the cluster, which decreases the cluster reliability and stability. Therefore, before allocating racks, take all aspects into consideration and properly set racks.

Rack Allocation Policies

NOTE

Physical rack: Indicates the real rack where the host resides.

Logical rack: Indicates the rack name of the host on FusionInsight Manager.

Policy 1: Each logical rack has nearly the same number of hosts.

Policy 2: The name of the logical rack of the host must comply with that of the physical rack to which the host belongs.

Policy 3: If there are only few hosts on a physical rack, combine this physical rack and other physical racks with few hosts into a logical rack, which complies with policy 1. Hosts in two equipment rooms cannot be placed in one logical rack. Otherwise, performance problems may be caused.

Policy 4: If there are lots of hosts on a physical rack, divide these hosts into multiple logical racks, which complies with policy 1. Hosts with great differences should not be placed in the same logical rack. Otherwise, the cluster reliability will be decreased.

Policy 5: You are advised to set **default** or other values for logical racks on the first layer, and the values in the same cluster must be consistent.

Policy 6: The number of hosts in each rack cannot be less than 3.

Policy 7: A cluster can contain at most 50 logical racks. If there are too many logical racks in a cluster, the maintenance is difficult.

Best Practice Example

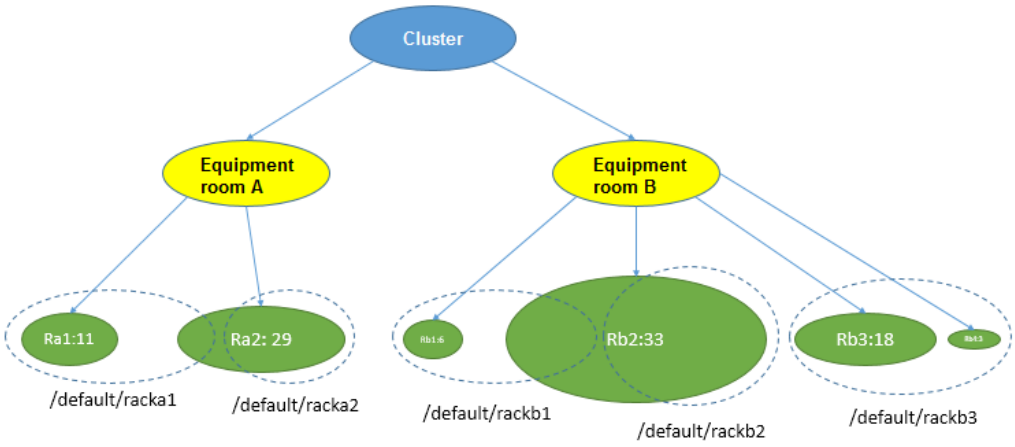
For example, there are 100 hosts in a cluster, 40 located in equipment room A and 60 located in equipment room B. In room A, there are 11 hosts on physical rack Ra1 and 29 hosts on physical rack Ra2. In room B, there are six hosts on physical rack Rb1, 33 hosts on physical rack Rb2, 18 hosts on physical rack Rb3, and three hosts on physical rack Rb4.

According to the rack allocation policy, each logical rack contains nearly the same number (for example, 20) of hosts. The allocation details are as follows:

- Logical rack /default/racka1: contains 11 hosts on physical rack Ra1 and nine hosts on physical rack Ra2
- Logical rack /default/racka2: contains the remaining 20 hosts (except the nine hosts of logical rack /default/racka1) on physical rack Ra2

- Logical rack /default/rackb1: contains six hosts on physical rack Rb1 and 13 hosts on physical rack Rb2
- Logical rack /default/rackb2: contains the remaining 20 hosts (except the 13 hosts of logical rack /default/rackb1) on physical rack Rb2
- Logical rack /default/rackb3: contains 18 hosts on physical rack Rb3 and three hosts on physical rack Rb4

Rack allocation example:



Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Hosts**.

Step 3 Select the check box of the target host.

Step 4 Select **Set Rack** from the **More** drop-down list.

- Set rack names in hierarchy based on the actual network topology. Separate racks from different layers using slashes (/).
- Rack naming rules are as follows: */level1/level2/...* The number of levels must be at least 1, and the name cannot be empty. A rack can contain letters, digits, and underscores (_) and cannot exceed 200 characters.

  For example, /default/rack0.
- If the hosts in the rack to be modified contain DataNode instances, ensure that the rack name levels of the hosts where all DataNode instances reside are the same. Otherwise, the configuration fails to be delivered.

Step 5 Click **OK**.

----End

10.4.2.4 Isolating a Host

Scenarios

If a host is abnormal or faulty and cannot provide services or affects the cluster performance, you can remove the host from the available node in the cluster temporarily so that the client can access other available nodes.

NOTE

Only non-management nodes can be isolated.

Impact on the System

- After a host is isolated, all role instances on the host will be stopped, and you cannot start, stop, or configure the host and all instances on the host.
- For some services, after a host is isolated, some instances on other nodes do not work, and the service configuration status may expire.
- After a host is isolated, statistics about the monitoring status and indicator data of the host hardware and instances on the host cannot be collected or displayed.
- Retain the default SSH port (22) of the target node. Otherwise, the task described in this section will fail.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Hosts**.

Step 3 Select the check box of the host to be isolated.

Step 4 Select **Isolate** from the **More** drop-down list.

In the displayed dialog box, enter the password of the current login user and click **OK**.

Step 5 In the displayed confirmation dialog box, select **I want to isolate the selected hosts and accept the consequences of possible service failures.** Click **OK**.

Wait until the message "Operation succeeded" is displayed, and click **Finish**.

Step 6 Log in to the isolated host as user **root** and run the **pkill -9 -u omm** command to stop the processes of user **omm** on the node. Then run the **ps -ef | grep'container' | grep '${BIGDATA_HOME}' | awk '{print $2}' | xargs -I' {}' kill -9' {}'** command to find and stop the container process.

Step 7 The host is successfully isolated and **Running Status** is **Isolated**.

If you have rectified the host exception or fault, cancel the isolation status of the host before using the host.

On the **Hosts** page, select the isolated host and choose **More > Cancel Isolation**.

NOTE

After the isolation is canceled, all role instances on the host are not started by default. To start role instances on the host, select the target host on the **Hosts** page and choose **More** > **Start All Instances**.

----End

10.4.2.5 Exporting Host Information

Scenarios

Administrators can export information about all hosts on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Hosts**.

Step 3 Specify the status of required hosts in the drop-down list box on the upper right corner, or click **Advanced Search** to specify hosts.

Step 4 Click **Export All** and select **TXT** or **CSV** for **Save As**. Then, click **OK**.

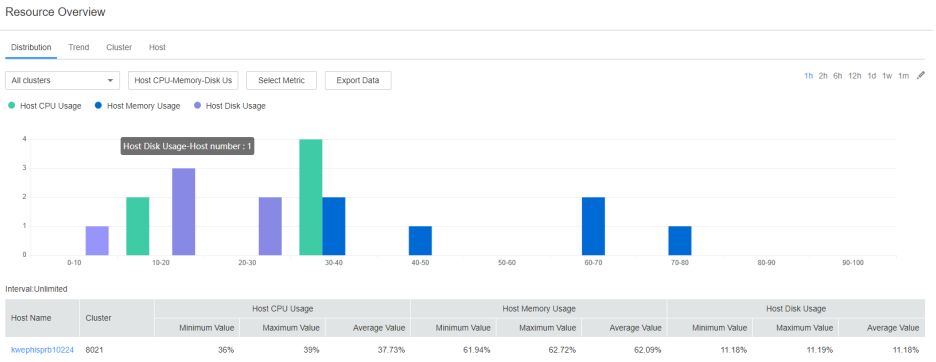
----End

10.4.3 Resource Overview

10.4.3.1 Distribution

Click **Hosts** > **Resource Overview** > **Distribution**, Figure 10-5 shows the interface for monitoring the cluster resource distribution. By default, the monitoring data of the latest one hour is displayed. You can click to customize a time range. The default time ranges are 1 hour, 2 hours, 6 hours, 12 hours, 1 day, 1 week, and 1 month.

Figure 10-5 Resource distribution overview



- Click **Select Metric** to customize the metrics to be viewed. Table 10-16 shows all the metrics. After selecting a metric, the distribution of hosts in the corresponding range is displayed.

- When you move the cursor to a color block, the number of hosts in the current CPU usage range is displayed, as shown in **Figure 10-5**. Click a column to display the hosts in the current range.
  - When you click the host name of a specified host, the page showing the host detailed information is displayed.
  - When you click **View Trends** of a specified host, the page showing the maximum, average, and minimum cluster values, and host values of the current metric are displayed. You cannot view the trends when the metric of the current cluster is **Host CPU-Memory-Disk Usage**.
- Click **Export Data** to export the metric maximum, minimum, and average values of all nodes in the current cluster in the selected time range.

Table 10-16 Metrics

Category	Metric
Process	• Number of Running Processes • Total Number of Processes • Total Number of omm Processes • Uninterruptible Sleep Process
Network Status	• Host Network Packet Collisions • Number of LAST_ACK States • Number of CLOSING States • Number of LISTENING States • Number of CLOSED States • Number of ESTABLISHED States • Number of SYN_RECV States • Number of TIME_WAITING States • Number of FIN_WAIT2 States • Number of FIN_WAIT1 States • Number of CLOSE_WAIT States • DNS Name Resolution Duration • TCP Ephemeral Port Usage • Host Network Packet Frame Errors
Network Reading	• Host Network Read Packets • Host Network Read Dropped Packets • Host Network Read Error Packets • Host Network Rx Speed
Disk	• Host Disk Write Speed • Host Used Disk • Host Free Disk • Host Disk Read Speed • Host Disk Usage

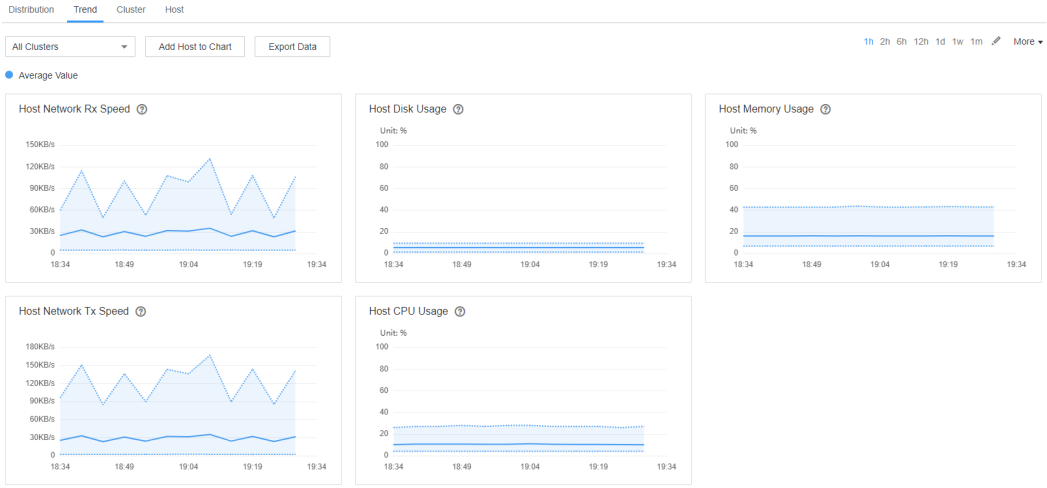
Category	Metric
Memory	• Free Memory • Cache Memory Size • Total Kernel Cache Memory Size • Shared Memory Size • Host Memory Usage • Used Memory
Network Writing	• Host Network Write Packets • Host Network Write Error Packets • Host Network Tx Speed • Host Network Write Dropped Packets
CPU	• CPU Usage of Processes Whose Priorities Have Been Changed • CPU Usage of User Space Processes • CPU Usage of Kernel Space Processes • Host CPU Usage • CPU Total Time • CPU Idle Time
Host Status	• Host File Handle Usage • Average OS Load in 1 Minute • Average OS Load in 5 Minutes • Average OS Load in 15 Minutes • Host PID Usage

10.4.3.2 Trend

Choose **Hosts > Resource Overview > Trend** to view the resource trend monitoring page of all clusters or a single cluster, as shown in **Figure 10-6**.

By default, the monitoring data of the latest one hour is displayed. You can click to customize a time range. The default time ranges are 1 hour, 2 hours, 6 hours, 12 hours, 1 day, 1 week, and 1 month. By default, the trend chart of each metric displays the maximum, minimum, and average values of the entire cluster.

Figure 10-6 Resource trend



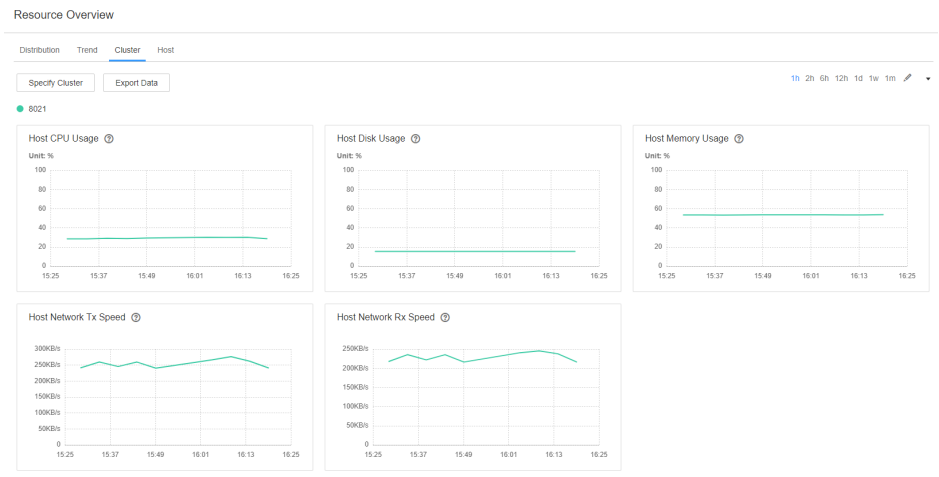
- Click **Add Host to Chart** to add trend lines of a host in the trend charts. A maximum of 12 hosts can be added.
- Choose > **Customize** to customize the metrics to be displayed on the page. For details about the metrics, see **Table 10-16** in section **Distribution**.
- Choose **Export Data** to export the maximum, minimum, and average values of all nodes in the cluster in the selected time range for all selected metrics.

10.4.3.3 Cluster

Choose **Hosts > Resource Overview > Cluster** to view the resource monitoring page of each cluster in FusionInsight Manager, as shown in **Figure 10-7**.

By default, the monitoring data of the latest one hour is displayed. You can click to customize a time range. The default time ranges are 1 hour, 2 hours, 6 hours, 12 hours, 1 day, 1 week, and 1 month.

Figure 10-7 Cluster resource overview



- Click **Specify Cluster** to customize the cluster to be displayed.

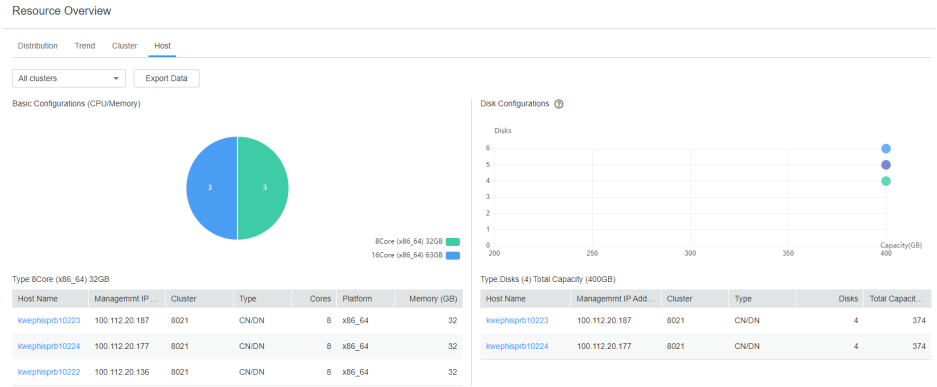
- Choose ▼ > **Customize** to customize the metrics to be displayed on the page. For details about the metrics, see **Table 10-16** in section **Distribution**.
- Choose **Export Data** to export the counter values of all selected counters under the selected time range for each cluster.

10.4.3.4 Host

Click **Hosts > Resource Overview > Host**, the host resources consist of basic resources (CPU/memory) and disk resources, as shown in **Figure 10-8**.

Click **Export Data** to export the configuration list of all hosts in the cluster, including the host name, management IP address, host type, number of cores, platform type, memory capacity, and disk size.

Figure 10-8 Host resource overview



Basic Configurations (CPU/Memory)

Move the cursor to the pie chart to display the information about the hardware configuration of each node in the current cluster. The information is displayed as follows: *Number of cores (platform type) Memory: Host quantity*.

Hosts that have different configurations are contained in different color blocks in a pie chart. You can click any color block to display the corresponding host list in the lower part of the page.

Disk Configurations

The horizontal axis indicates the disk capacity (including the OS disk) of nodes, and the vertical axis indicates the number of logical disks (including the OS disk) of nodes.

When you place the cursor on a dot, the information about the disks in the current configuration state, including the number of disks, total capacity, and number of hosts, is displayed.

Click a dot to display the disks of this configuration in the lower part of the page.

10.5 O&M

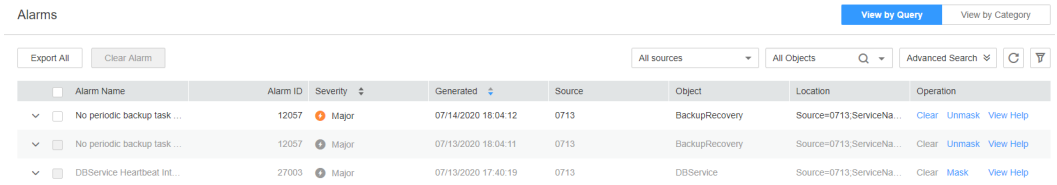
10.5.1 Alarms

10.5.1.1 Overview of Alarms and Events

Alarms

Log in to FusionInsight Manager and choose **O&M** > **Alarm** > **Alarms**. **Figure 10-9** is displayed. You can view alarm information reported by clusters in FusionInsight Manager, including the alarm name, ID, severity, and generation time. By default, the latest ten alarms are displayed on each page.

Figure 10-9 FusionInsight Manager alarm management



Detailed alarm parameters

Click ⌄ on the left of a specified alarm to expand the alarm parameters. **Table 10-17** describes the parameters.

Table 10-17 Alarm parameters

Alarm Parameter	Description
Alarm ID	Indicates the alarm ID.
Alarm Name	Indicates the alarm information name.
Alarm Severity	There are four levels: critical, major, minor, and warning.
Source	Cluster name.
Cleared	Indicates the time when an alarm is cleared. If the alarm is not cleared, -- is displayed.
Object	Indicates the services, processes, or modules that triggers an alarm.
Automatic Clearance	The alarm can be automatically cleared after the fault is rectified.
Alarm Status	Indicates the status of the alarm. Manually Cleared Indicates the current alarm status, including automatic clearance, manual clearance, and uncleared.
Generated	Indicates the time when the alarm is generated

Alarm Parameter	Description
Alarm Cause	Indicates the possible cause of an alarm.
Serial Number	Indicates the number of alarms generated by the system.
Additional Information	Indicates the error information.
Location	Indicates the detailed information for locating the alarm, which includes the following: ● Source: identifies the cluster for which the alarm is generated. ● ServiceName: identifies the service for which the alarm is generated. ● RoleName: identifies the role for which the alarm is generated. ● HostName: identifies the host for which the alarm is generated.

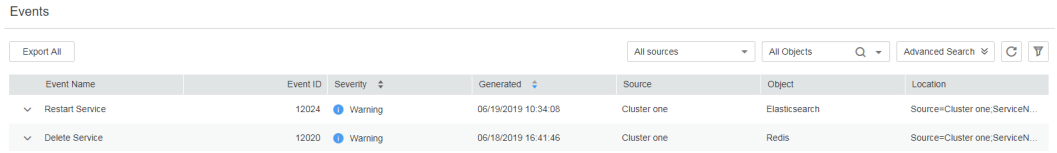
Manage Alarms:

- Click **Export All** to export all alarm details.
- If multiple alarms have been handled, you can select one or more alarms to be cleared and click **Clear Alarm** to clear the alarms in batches. A maximum of 300 alarms can be cleared in each batch.
- Click to manually refresh the current page. Click to filter alarms displayed on the page.
- You are allowed to filter specified alarms by object or cluster.
- Click **Advanced Search**. An area where you can search for alarms is displayed. You can search for alarms by alarm ID, alarm name, alarm type, start time, and end time. Click **Search** to display filtered alarms. After you click **Advanced Search** again, the number of entered search criteria is displayed.
- You can click **Clear**, **Masking**, or **Help** to perform corresponding operations on an alarm.
- If there are a large number of alarms, you can click **View by Category**. The system classifies uncleared alarms by alarm ID. After the alarm is classified, click the number of uncleared alarms next to the alarm name to view the alarm details.

Events

Log in to FusionInsight Manager and choose **O&M** > **Alarm** > **Events**. On the displayed page, you can view the information about all events in the cluster, including the alarm name, ID, severity, generation time, object, and location. By default, the latest 10 events are displayed on each page.

Figure 10-10 FusionInsight Manager event management



Click ⌄ on the left of a specified event. The related event parameters are displayed, as shown in **Table 10-18**.

Table 10-18 Event parameters

Parameter	Description
Event ID	Indicates the event ID.
Event Name	Indicates the event name.
Severity	Indicates the event severity. There are four levels: Critical, major, minor, and warning.
Generated	Indicates the time when an event occurs.
Object	Indicates the possible cause of an event.
Serial Number	Indicates the number of events generated in the system.
Location	Indicates the detailed information for locating the event, which includes the following: • Source: identifies the cluster for which the event is generated. • ServiceName: identifies the service for which the event is generated. • RoleName: identifies the role for which the event is generated. • HostName: identifies the host for which the event is generated.
Additional Information	Indicates the error information.
Event Cause	Indicates the possible cause of an event.
Source	Cluster name.

Manage Events:

- Click **Export All** to export all event details.
- Click ↻ to manually refresh the current page. Click ▽ to filter events displayed on the page.
- You are allowed to filter specified events by object or cluster.

- Click **Advanced Search**. An area where you can search for events is displayed. You can search for events by event ID, severity, event name, start time, and end time.

10.5.1.2 Configuring the Threshold

Scenarios

You can configure monitoring indicator thresholds to monitor the health status of indicators on FusionInsight Manager. If abnormal data occurs and the preset conditions are met, the system triggers an alarm and displays the alarm information on the alarm page.

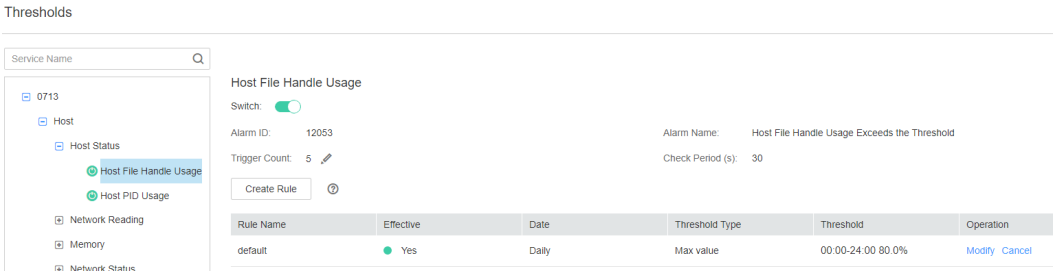
Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Alarm** > **Thresholds**.

Step 3 Select a monitoring indicator for a specified host or service in the cluster.

Figure 10-11 Configuring indicator thresholds



For example, after selecting **Host Memory Usage**, the information about this indicator threshold is displayed.

- If the alarm sending switch is displayed as , an alarm is triggered if the alarm threshold is reached.
- The alarm ID and alarm name contain the alarm information that is triggered by the threshold:
- FusionInsight Manager checks whether the value of each monitored indicator reaches the threshold. If the number of consecutive check times is equal to the value of **Trigger Count**, and the threshold is not reached in these checks, the system sends an alarm.
- The value can be customized. **Check Period (s)** indicates the interval for the system to check monitoring indicators.
- Rules for triggering an alarm.

Step 4 Click **Create Rule** to add rules used for monitoring indicators.

Table 10-19 Monitoring indicator rule parameters

Parameter	Value	Description
Rule Name	CPU_MAX (example value)	Name of a rule.
Alarm Severity	• Critical • Major • Minor • Warning	Alarm Severity • Critical • Major • Minor • Warning
Threshold Type	• Max value • Min value	You can select the maximum or minimum value of an indicator. Setting this parameter to **Max value**, the system generates an alarm when the actual value of the indicator is greater than the threshold. Setting this parameter to **Min value**, the system generates an alarm when the actual value of the indicator is less than the threshold.
Date	• Daily • Weekly • Others	This parameter is used to set the date when the rule takes effect.
Add Date	09-30	This parameter is available only when **Date** is set to **Others**. You can set the date when the rule takes effect. Multiple options are available.
Thresholds	Start and End Time: 00:00 to 08:30	This parameter is used to set the time range when the rule takes effect.
	Threshold: 10	Specifies the threshold of the rule monitoring indicator.

NOTE

For the last parameter in the table, you can click or to add or delete multiple start and end time or alarm indicator thresholds.

Step 5 Click **OK** to save the rules.

Step 6 Locate the row that contains an added rule, and click **Apply** in the **Operation** column. The value of **Effective** for this rule changes as **Yes**.

You can apply a new rule only after clicking **Cancel**.

----End

Monitoring Indicator Reference

FusionInsight Manager alarm monitoring indicators are categorized into node information indicators and cluster service indicators. **Table 10-20** describes the indicators whose thresholds can be configured on nodes.

Table 10-20 Monitoring indicators on each node

Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
CPU	Host CPU Usage	This indicator reflects the computing and control capabilities of the current cluster in a measurement period. By observing the indicator value, you can better understand the overall resource usage of the cluster.	90.0%
Disk	Disk Usage	Indicates the disk usage of a host.	90.0%
	Disk Inode Usage	Indicates the disk inode usage in a measurement period.	80.0%
Memory	Host Memory Usage	Indicates the average memory usage at the current time.	90.0%

Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
Host Status	Host File Handle Usage	Indicates the usage of file handles of the host in a measurement period.	80.0%
	Host PID Usage	Indicates the PID usage of a host.	90%
Network Status	TCP Ephemeral Port Usage	Indicates the usage of temporary TCP ports of the host in a measurement period.	80.0%
Network Reading	Read Packet Error Rate	Indicates the read packet error rate of the network interface on the host in a measurement period.	0.5%
	Read Packet Dropped Rate	Indicates the read packet dropped rate of the network interface on the host in a measurement period.	0.5%
	Read Throughput Rate	Indicates the average read throughput (at MAC layer) of the network interface in a measurement period.	80%
Network Writing	Write Packet Error Rate	Indicates the write packet error rate of the network interface on the host in a measurement period.	0.5%

Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
	Write Packet Dropped Rate	Indicates the write packet dropped rate of the network interface on the host in a measurement period.	0.5%
	Write Throughput Rate	Indicates the average write throughput (at MAC layer) of the network interface in a measurement period.	80%
Process	Uninterruptible Sleep Process	Indicates the number of D state processes on the host in a measurement period.	0
	omm Process Usage	Indicates the usage of the omm process within a measurement period.	90

Table 10-21 Cluster service indicators

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
DBService	Database	Database Connections Usage	Indicates the usage of the number of database connections.	90%
		Disk Space Usage of the Data Directory	Disk space usage of the data directory.	80%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
Flume	Agent	Heap Memory Usage Calculate	Indicates the Flume heap memory usage.	95.0%
		Flume Direct Memory Usage Statistics	Indicates the Flume direct memory usage.	80.0%
		Flume Non-heap Memory Usage	Indicates the Flume non-heap memory usage.	80.0%
		Total GC duration of Flume process	Indicates the Flume total GC time.	12000ms
HBase	GC	GC time for old generation	Indicates the total GC time of RegionServer.	5000ms
		GC time for old generation	Indicates the total GC time of HMaster.	5000ms
	CPU and Memory	RegionServer Direct Memory Usage Statistics	Indicates the RegionServer Reg direct memory usage.	90%
		RegionServer Heap Memory Usage Statistics	Indicates the RegionServer heap memory usage.	90%
		HMaster Direct Memory Usage	Indicates the HMaster direct memory usage.	90%
		HMaster Heap Memory Usage Statistics	Indicates the HMaster heap memory usage.	90%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
	Service	Regions	Indicates the number of regions of a RegionServer.	2000
		Region in transaction count over threshold	Number of regions that are in the RIT state and reach the threshold duration.	1
	Replication	Replication sync failed times	Indicates the number of times that DR data fails to be synchronized.	1
	Queue	Compaction Queue Size	Compaction queue size.	100
HDFS	File and Block	Lost Blocks	Number of missing copy blocks in the HDFS file system.	0
		Blocks Under Replicated	Total number of blocks that need to be replicated by the NameNode.	1000
	RPC	Average Time of Active NameNode RPC Processing	Indicates the average RPC processing time.	100ms
		Average Time of Active NameNode RPC Queuing	Indicates the average RPC queuing time.	200ms
	Disk	Disk Usage	Indicates the HDFS disk usage.	80%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		Percentage of DataNode Capacity	Indicates the disk usage of DataNodes in the HDFS.	80%
		Percentage of Reserved Space for Replicas of Unused Space	Indicates the percentage of the reserved disk space of all the copies to the total unused disk space of DataNodes.	90%
	Resource	Faulty DataNodes	Indicates the number of faulty DataNodes.	3
		NameNode Non Heap Memory Usage Statistics	Indicates the percentage of NameNode non-heap memory usage.	90%
		NameNode Direct Memory Usage Statistics	Indicates the percentage of direct memory used by NameNodes.	90%
		NameNode Heap Memory Usage Statistics	Indicates the percentage of NameNode non-heap memory usage.	95%
		DataNode Non Heap Memory Usage Statistics	Indicates the percentage of DataNode non-heap memory usage.	90%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		DataNode Direct Memory Usage Statistics	Indicates the percentage of direct memory used by DataNodes.	90%
		DataNode Heap Memory Usage Statistics	Indicates the percentage of DataNode non-heap memory usage.	95%
	Garbage Collection	GC Time	Indicates the Garbage collection (GC) duration of NameNodes per minute.	12000ms
		GC Time	Indicates the GC duration of DataNodes per minute.	12000ms
Hive	HQL	Percentage of HQL Statements That Are Executed Successfully by Hive	Indicates the percentage of HQL statements that are executed successfully by Hive.	90.0%
	Background	Background Thread Usage	Indicates the percentage of Background thread usage.	90%
	GC	Total GC Time in Milliseconds	Indicates the total GC time of MetaStore.	12000ms
		Total GC Time in Milliseconds	Indicates the total GC time of HiveServer.	12000ms

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
	Capacity	Percentage of HDFS Space Used by Hive to the Available Space	Indicates the percentage of HDFS space used by Hive to the available space.	85.0%
	CPU and Memory	MetaStore Direct Memory Usage Statistics	Indicates the MetaStore direct memory usage.	95%
		MetaStore Non-Heap Memory Usage Statistics	Indicates the MetaStore non-heap memory usage.	95%
		MetaStore Heap Memory Usage Statistics	Indicates the MetaStore heap memory usage.	95%
		HiveServer Direct Memory Usage Statistics	Indicates the HiveServer direct memory usage.	95%
		HiveServer Non-Heap Memory Usage Statistics	Indicates the HiveServer non-heap memory usage.	95%
		HiveServer Heap Memory Usage Statistics	Indicates the HiveServer heap memory usage.	95%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
	Session	Percentage of Sessions Connected to the HiveServer to Maximum Number of Sessions Allowed by the HiveServer	Indicates the percentage of the number of sessions connected to the HiveServer to the maximum number of sessions allowed by the HiveServer.	90.0%
Kafka	Partition	Percentage of Partitions That Are Not Completely Synchronized	Indicates the percentage of partitions that are not completely synchronized to total partitions.	50%
	Other	Unavailable Partition Percentage	Disk usage of the disk where the Broker data directory is located.	40%
		User Connection Usage on Broker	User connection usage on the broker.	80%
	Disk	Broker Disk Usage	Indicates the disk usage of the disk where the Broker data directory is located.	80%
	Process	Broker GC Duration per Minute	Indicates the GC duration of the Broker process per minute.	12000ms

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		Heap Memory Usage of Kafka	Indicates the Kafka heap memory usage.	95%
		Kafka Direct Memory Usage	Indicates the Kafka direct memory usage.	95%
Loader	Memory	Heap Memory Usage Calculate	Indicates the Loader heap memory usage.	95%
		Loader Direct Memory Usage Statistics	Indicates the Loader direct memory usage.	80.0%
		Non heap Memory Usage Calculate	Indicates the Loader non-heap memory usage.	80%
	GC	Total GC time in milliseconds	Indicates the total GC time of Loader.	12000ms
MapReduce	Garbage Collection	GC Time	Indicates the GC time.	12000ms
	Resource	JobHistoryServer Direct Memory Usage Statistics	Indicates the JobHistoryServer direct memory usage.	90%
		JobHistoryServer Non Heap Memory Usage Statistics	Indicates the JobHistoryServer non-heap memory usage.	90%
		JobHistoryServer Heap Memory Usage Statistics	Indicates the JobHistoryServer non-heap memory usage.	95%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
Oozie	Memory	Heap Memory Usage Calculate	Indicates the Oozie heap memory usage.	95.0%
		Oozie Direct Buffer Resource Percentage	Indicates the Oozie direct memory usage.	80.0%
		Non Heap Memory Usage Calculate	Indicates the Oozie non-heap memory usage.	80%
	GC	Total GC duration of Oozie process	Indicates the Oozie total GC time.	12000ms
Spark2x	Memory	JDBCServer2x Heap Memory Usage Statistics	Indicates the JDBCServer2x heap memory usage.	95%
		JDBCServer2x Direct Memory Usage Statistics	Indicates the JDBCServer2x direct memory usage.	95%
		JDBCServer2x Non-Heap Memory Usage Statistics	Indicates the JDBCServer2x non-heap memory usage.	95%
		JobHistory2x Direct Memory Usage Statistics	Indicates the JobHistory2x direct memory usage.	95%
		JobHistory2x Non-Heap Memory Usage Statistics	Indicates the JobHistory2x non-heap memory usage.	95%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		JobHistory2x Heap Memory Usage Statistics	Indicates the JobHistory2x heap memory usage.	95%
		IndexServer2x Direct Memory Usage Statistics	Indicates the IndexServer2x direct memory usage.	95%
		IndexServer2x Heap Memory Usage Statistics	ndicates the IndexServer2x heap memory usage.	95%
		IndexServer2x Non-Heap Memory Usage Statistics	Indicates the IndexServer2x non-heap memory usage.	95%
	GC number	Full GC Number of JDBCServer2x	Indicates the total GC number of JDBCServer2x.	12
		Full GC Number of JobHistory2x	Indicates the total GC number of JobHistory2x.	12
		Full GC Number of IndexServer2x	Indicates the total GC number of IndexServer2x.	12
	GC Time	Total GC time in milliseconds	Indicates the total GC time of JDBCServer2x.	12000ms
		Total GC time in milliseconds	Indicates the total GC time of JobHistory2x.	12000ms

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		Total GC time in milliseconds	Indicates the total GC time of IndexServer2x.	12000ms
Storm	Cluster	Number of Available Supervisors	Indicates the number of available Supervisor processes in the cluster in a measurement period.	1
		Slot Usage	Indicates the slot usage in the cluster in a measurement period.	80.0%
	Nimbus	Heap Memory Usage Calculate	Indicates the Nimbus heap memory usage.	80%
Yarn	Resource	NodeManager Direct Memory Usage Statistics	Indicates the percentage of direct memory used by NodeManagers.	90%
		NodeManager Heap Memory Usage Statistics	Indicates the percentage of NodeManager heap memory usage.	95%
		NodeManager Non Heap Memory Usage Statistics	Indicates the percentage of NodeManager non-heap memory usage.	90%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		ResourceManager Direct Memory Usage Statistics	Indicates the Kafka direct memory usage.	90%
		ResourceManager Heap Memory Usage Statistics	Indicates the ResourceManager heap memory usage.	95%
		ResourceManager Non Heap Memory Usage Statistics	Indicates the ResourceManager non-heap memory usage.	90%
	CPU and Memory	Pending Memory	Pending memory capacity.	83886080MB
	Other	Failed Applications of root queue	Number of failed tasks in the **root** queue.	50
		Terminated Applications of root queue	Number of killed tasks in the **root** queue.	50
	Garbage collection	GC Time	Indicates the GC duration of NodeManager per minute.	12000ms
		GC Time	Indicates the GC duration of ResourceManager per minute.	12000ms
	Application	Pending Applications	Pending tasks.	60

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
ZooKeeper	Connection	ZooKeeper Connections Usage	Indicates the percentage of the used connections to the total connections of ZooKeeper.	80%
	CPU and Memory	Heap Memory Usage Calculate	Indicates the ZooKeeper direct memory usage.	95%
		Direct Memory Usage Calculate	Indicates the ZooKeeper heap memory usage.	80%
	GC	ZooKeeper GC Duration per Minute	Indicates the GC time of ZooKeeper every minute.	12000ms
meta	OBS Meta data Operations	Average Time for Calling the OBS Metadata API	Average time for calling the OBS metadata APIs.	500ms
		Success Rate for Calling the OBS Metadata API	Success rate of calling the OBS metadata APIs	99.0%
	OBS data write operation	Success Rate for Calling the OBS Write API	Success rate of calling the OBS data write APIs.	99.0%
	OBS data read operation	Success Rate for Calling the OBS Data Read API	Success rate of calling the OBS data read operation APIs.	99.0%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
Ranger	GC	UserSync GC Duration	UserSync garbage collection (GC) duration.	12000ms
		RangerAdmin GC Duration	RangerAdmin garbage collection (GC) duration.	12000ms
		TagSync GC Duration	TagSync garbage collection (GC) duration.	12000ms
	CPU and Memory	UserSync Non-Heap Memory Usage	UserSync non-heap memory usage in percentage.	80.0%
		UserSync Direct Memory Usage	UserSync direct memory usage in percentage.	80.0%
		UserSync Heap Memory Usage	UserSync heap memory usage in percentage.	95.0%
		RangerAdmin Non-Heap Memory Usage	RangerAdmin non-heap memory usage.	80.0%
		RangerAdmin Heap Memory Usage	RangerAdmin heap memory usage in percentage.	95.0%
		RangerAdmin Direct Memory Usage	RangerAdmin direct memory usage.	80.0%

Service	Monitoring Indicator Group Name	Indicator Name	Description	Default Threshold
		TagSync Direct Memory Usage	TagSync direct memory usage in percentage.	80.0%
		TagSync Non-Heap Memory Usage	TagSync non-heap memory usage in percentage.	80.0%
		TagSync Heap Memory Usage	TagSync heap memory usage in percentage.	95.0%
ClickHouse	Cluster Quota	Clickhouse service quantity quota usage in ZooKeeper	Quota of the ZooKeeper nodes used by the ClickHouse service.	90%
		Capacity quota usage of the Clickhouse service in ZooKeeper	Capacity quota of ZooKeeper directory used by the ClickHouse service.	90%
IoTDB	GC	IoTDBServer GC Duration	IoTDBServer garbage collection (GC) duration.	12000ms
	CPU and Memory	IoTDBServer Heap Memory Usage	IoTDBServer heap memory usage in percentage.	90%
		IoTDBServer Direct Memory Usage	IoTDBServer direct memory usage in percentage.	90%

10.5.1.3 Configuring the Alarm Masking Status

Scenarios

If you do not want FusionInsight Manager to report specified alarms in the following scenarios, you can manually mask the alarms.

- Some unimportant alarms and minor alarms need to be masked.
- When a third-party product is integrated with FusionInsight, some alarms of the product are duplicated with the alarms of FusionInsight and need to be masked.
- When the deployment environment is special, certain alarms may be falsely reported and need to be masked.

After an alarm is masked, new alarms with the same ID as the alarm are neither displayed on the **Alarm** page nor counted. The reported alarms are still displayed.

Procedure

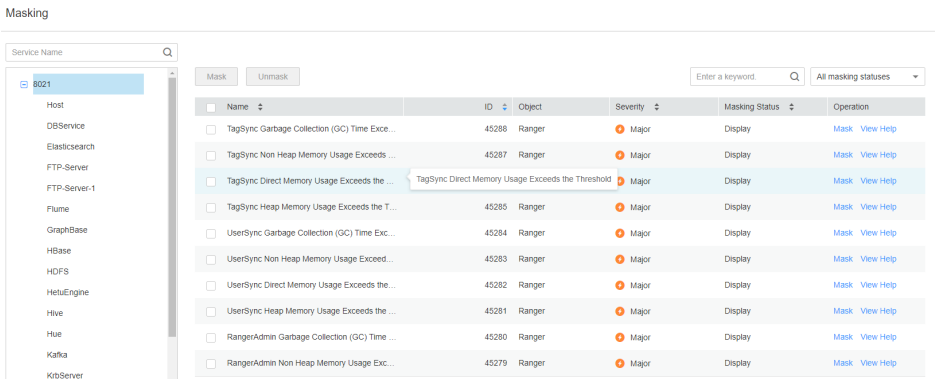
Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Alarm** > **Masking**.

Step 3 In the **Masking** area, select the specified service or module.

Step 4 Select an alarm from the alarm list.

Figure 10-12 Masking an alarm



The information about the alarm is displayed, including the alarm name, ID, severity, masking status, and operations can be performed on the alarm.

- The masking status includes **Display** and **Mask**.
- Operations include **Mask** and **View Help**.

NOTE

You can filter specified alarms based on the masking status and alarm severity.

Step 5 Set the masking status for an alarm:

- Click **Mask**. In the displayed dialog box, click **OK** to change the alarm masking status to **Mask**.

- Click **Cancel Masking**. In the displayed dialog box, click **OK** to change the masking status of the alarm to **Display**.

----End

10.5.2 Log

10.5.2.1 Online Log Searching

Scenarios

FusionInsight Manager supports online search and displays component logs for log viewing scenarios, such as fault locating.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M > Log > Online Search**.

Step 3 Set the parameters listed in **Table 10-22** based on the logs to be queried. You can select a log generation time range, the default time ranges are 0.5 hour, 1 hour, 2 hours, 6 hours, 12 hours, 1 day, 1 week, and 1 month, or click to customize **Start Date** and **End Data**.

Table 10-22 Log search parameters

Parameter	Description
Search Content	Specifies the keywords or regular expressions to be searched for.
Service	Specifies the service or module for which you want to query logs.
File	Specifies the log file to be searched when only one role is selected.
Lowest Log Level	Specifies the lowest level of the logs to be queried. After selecting a level, logs of this level and higher are displayed. The log levels from low to high are as follows: TRACE, DEBUG, INFO, WARN, ERROR, and FATAL

Parameter	Description
Host Scope	• You can click to select the hosts. • Enter the host name of the node for which you want to query logs or the IP address of the management plane. • Use commas (,) to separate IP addresses. For example, **192.168.10.10**,**192.168.10.11**. • Use hyphens (-) to indicate an IP address segment if the IP addresses are consecutive. For example, **192.168.10.[10-20]**. • Use hyphens (-) to indicate an IP address segment if the IP addresses are consecutive, and use commas (,) to separate IP address segments. For example, **192.168.10.[10-20,30-40]**. **NOTE** • If this parameter is not specified, all hosts are selected by default. • A maximum of 10 expressions can be entered at a time. • A maximum of 2000 hosts can be matched for all entered expressions at a time.
Advanced Configurations	• **Max Quantity**: specifies the maximum number of logs that can be displayed at a time. If the number of queried logs exceeds the preset value, the earliest logs will be ignored. If this parameter is not set, the maximum number of logs that can be displayed at a time is not limited. • **Timeout Duration**: specifies the log query timeout duration. This parameter is used to limit the maximum log query time on each node. When the query times out, the query is stopped and the query results are still displayed.

Step 4 Click **Search** to start the search. **Table 10-23**describes the fields in the query result.

Table 10-23 Search results

Parameter	Description
Time	Specifies the time when a piece of log is generated.
Source Cluster	Cluster where logs are generated.
Host Name	Specifies the host name of the node where the log file that records the line of log is located.

Parameter	Description
Location	Specifies the path of the log file that records the line of log. Click the location information to go to the online log browsing page. By default, 100 lines of logs before and 100 lines after the line of log are displayed. You can click **Load More** on the top or bottom of the page to view more logs. Click **Download** to download the log file to the local PC.
Line No.	Specifies the line number of a line of log in the log file.
Level	Specifies the log level.
Log	Specifies the log content.

NOTE

You can click **Stop** to forcibly stop the retrieval. The retrieved results are displayed in the list.

Step 5 Click **Filter** to filter the logs displayed on the page. **Table 10-24** lists the fields that you can use to filter logs. After setting these parameters, click **Filter** to search for logs meeting the search criteria. You can click **Reset** to clear the entered information.

Table 10-24 Filter

Parameter	Description
Keywords	Specifies the keywords of the logs to be searched for.
Host Name	Specifies the name of the host to be searched for.
Location	Specifies the path of the log file to be searched for.
Started	Specifies the start time for logs to be searched for.
Completed	Specifies the end time for logs to be searched for.
Source Cluster	Specifies the cluster of the logs to be searched for.

----End

10.5.2.2 Log Downloadind

Scenarios

FusionInsight Manager allows you to export logs generated by all instances of each service role in batches. You do not need to manually log in to a specified node to obtain the logs.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Log > Download**.

Step 3 Select a log downloading range:

1. **Service**: Click ... to set a service for **Service**.
2. **Host**: Set the IP address of the host where the service is deployed. You can also click ... to select the host.
3. Click  in the upper right corner to select the corresponding **Start Date** and **End Date**.

Step 4 Click **Download**.

The downloaded log package contains the topology information of the corresponding start time and end time, facilitating locating.

The topology file is named in the format of **topo_<***Topology change time***>.txt.** The file contains the node IP address, node host name, and service instances installed on the node. (The OMS node is identified by Manager:Manager.)

The following shows an example:

```
192.168.204.124|suse-124|
DBService:DBServer;KrbClient:KerberosClient;LdapClient:SlapdClient;LdapServer:SlapdServer;Manager:Manager;meta:meta
```

----End

10.5.3 Perform a Health Check

10.5.3.1 Viewing a Health Check Task

Scenarios

Administrators can view all health check tasks in the health check management center to check whether the cluster is affected after the modification.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Health Check**.

By default, all the saved check reports are displayed in a list, as listed in the following table.

Table 10-25 Health check report records

Item	Description
Checked Object	Indicates the object being checked, open the drop-down menu to view the details.
Status	Indicates the check result status, including **No problems found**, **Problems found**, and **Checking**.
Check Type	Indicates the entity on which the health check is performed, including four dimensions: **System**, **Cluster**, **Host**, **Service**, and **OMS**. By default, a health check on the cluster dimension contains all checks items.
Start Mode	Indicates whether the health check is automatically triggered or manually executed.
Started	Indicates the start time of the check.
Completed	Indicates the end time of the check.
Operation	You can export the health check report and view the help information.

NOTE

- In the upper pane, you can filter specified health check records by check object and result status.
- If the cluster checked, you can click **Help** in the **Checked Object** drop-down list box.
- During the health check, the system collects the recent historical data instead of the real-time monitoring data of object indicators. Therefore, the check is delayed.

----End

10.5.3.2 Managing Health Check Reports

Scenarios

You can manage all saved health check reports on FusionInsight Manager. That is, you can download or delete historical health check reports.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Health Check**.

Step 3 Locate the row that contains a target health check report, click **Export Report**, and download the report file.

----End

10.5.3.3 Modifying Health Check Configuration

Scenarios

Administrators can enable automatic health check to reduce manual operation time. By default, the automatic health check checks the entire cluster.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Health Check** > **Configuration**.

Periodic Health Check indicates whether to enable automatic health check. Selecting **Enable** to enable the automatic health check, and selecting **Disable** to disable the function.

Set the health check period to **Daily**, **Weekly**, or **Monthly** as required.

Step 3 Click **OK** to save the configurations.

----End

10.5.4 Configuring Backup and Backup Restoration

10.5.4.1 Creating a Backup Task

Scenarios

You can create backup tasks on FusionInsight Manager. Executing backup tasks backs up related data.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Backup and Restoration** > **Backup Management** > **Create**.

Step 3 Set **Backup Object** to **OMS** or the cluster whose data needs to be backed up.

Step 4 Enter a task name in the **Name** text box.

Step 5 Set **Mode** to **Periodic** or **Manual** as required.

Table 10-26 Backup types

Type	Parameter	Description
Periodic	Started	Indicates the time when a periodic backup task is started for the first time.
	Period	Indicates the interval between the time when a task is executed last time and that when the task is started next time. The unit can be hour or day.
	Backup Policy	The following policies can be selected: • Full backup at the first time and subsequent incremental backup • Full backup every time • Full backup once every n times
Manual	N/A	You need to manually execute the task to back up data.

Step 6 Set required parameters in the **Configuration** area.

- Metadata and service data can be backed up.
- For details about how to back up data of different components, see **Backup and Recovery Management**.

Step 7 Click **OK** to save the configurations.

Step 8 In the backup task list, you can view the created backup task.

Locate the row that contains the target backup task, choose **More** > **Back Up Now** in the **Operation** column to execute the task immediately.

----End

10.5.4.2 Creating a Backup Restoration Task

Scenarios

You can create a backup restoration task on FusionInsight Manager. After the restoration task is executed, the specified backup data is restored to the cluster.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Backup and Restoration** > **Restoration Management**. On the displayed page, click **Create**.

Step 3 Set **Recovery Object** to **OMS** or the cluster whose data needs to be restored.

Step 4 Enter a task name in the **Task Name** text box.

Step 5 Set the required parameters in the **Recovery Configuration** area.

- Metadata and service data can be restored.
- For details about how to how to restore data of different components, see **Backup and Recovery Management**.

Step 6 Click **OK** to save the configurations.

Step 7 In the restoration task list, you can view the created restoration tasks.

Locate the row that contains the target backup restoration task, click **Start** in the **Operation** column to execute the restoration task immediately.

----End

10.5.4.3 Managing Backup and Backup Restoration Tasks

Scenarios

You can also maintain and manage backup restoration tasks on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **O&M** > **Backup and Restoration** > **Backup Management** or **O&M** > **Backup and Restoration** > **Restoration Management**.

Step 3 In the **Operation** column of the specified task in the task list, select the operation to be performed.

Table 10-27 Maintenance and management operations

Operation Entry	Description
Config	Modify parameters for the backup task.
Recover	After some service data is successfully backed up, you can use this function to quickly restore data.
More > **Back Up Now**	Perform this operation to execute the backup task immediately.
More > **Stop**	Perform this operation to stop a running task.
More > **Delete** or **Delete**	This operation is used to delete tasks.
More > **Suspend**	Perform this operation to disable the automatic backup task function.
More > **Resume**	Perform this operation to enable the automatic backup task function.
More > **View History** or **View History**	Perform this operation to switch to the task run log page to view the task running details and backup path.

Operation Entry	Description
View	Perform this operation to check the parameter settings of the restoration task.
Start	Perform this operation to run the restoration task.

----End

10.6 Audit

10.6.1 Overview

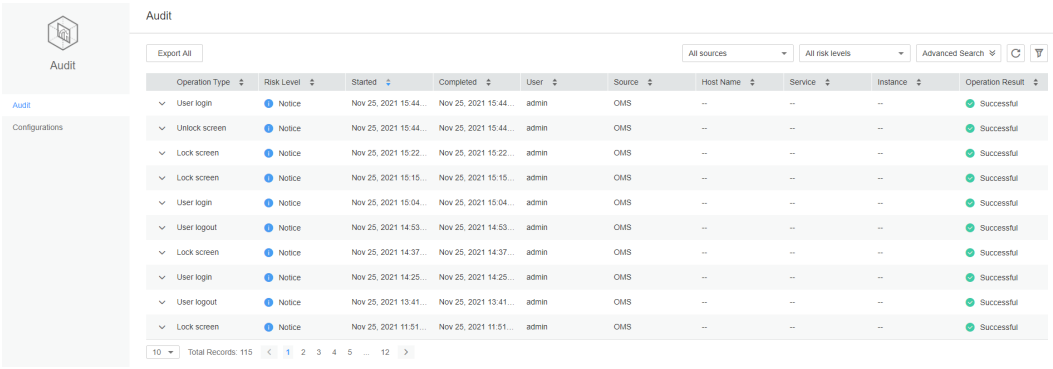
Scenario

The **Audit** page displays the user operations on Manager. On this page, administrators can view historical user operations on Manager. For details about the audit information, see **Audit Logs**.

Overview

Log in to FusionInsight Manager and choose **Audit**. **Figure 10-13** shows the audit information, including the operation type, risk level, start time, end time, user, host name, service, instance, and operation result.

Figure 10-13 Audit information list



- You can select audit logs at the **Major**, **Minor**, or **Notice** level from the **All risk levels** drop-down list.
- In **Advanced Search**, you can set filter criteria to query audit logs.
  a. You can query audit logs by user management, cluster, service, and health in the **Operation Type** column.
  b. In the **Service** column, you can select a service to query corresponding audit logs.

NOTE

You can select -- to search for audit logs using all other search criteria except services.

c. You can query audit logs by operation result, such as, **Successful**, **Failed**, or **Unknown**, as shown in the following figure.

- You can click to manually refresh the current page or click to filter the columns displayed in the page.
- Click **Export All** to export all audit information at a time. The audit information can be exported in **TXT** or **CSV** format.

10.6.2 Configuring Audit Log Dumping

Scenarios

The audit logs of FusionInsight Manager are stored in the database by default. If the audit logs are retained for a long time, the disk space of the data directory may be insufficient. To store audit logs to another archive server, administrators can set the required dump parameters to automatically dump these logs. This facilitates the management of audit logs.

If you do not configure the audit log dumping, the system automatically saves the audit logs to a file when the number of audit logs reaches 100,000 pieces. The save path is **${BIGDATA_DATA_HOME} /dbdata_om/dumpData/iam/operatelog** on the active management node. The file name format is **OperateLog_store**_*YY_MM_DD_HH_MM_SS*.**csv**. The maximum number of historical audit log files is 50.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Audit** > **Configurations**.

Step 3 Click the switch on the right of **Dump Audit Log**.

By default, **Dump Audit Log** is disabled. If this parameter is set to , the function is enabled.

Step 4 Set the dump parameters based on information provided in **Table 10-28**.

Table 10-28 Audit log dump parameters

Parameter	Description	Value
SFTP IP Mode	IP address mode. The value can be **IPv4** or **IPv6**.	**IPv4**
SFTP IP	Specifies the SFTP server for storing dumped audit logs. This parameter is mandatory. You are advised to use the SFTP service based on SSH v2. Otherwise, security risks exist.	**192.168.10.51** (example value)

Parameter	Description	Value
SFTP Port	Specifies the port of the SFTP server for storing dumped audit logs. This parameter is mandatory.	**22** (example value)
Save Path	Specifies the path for storing audit logs on the SFTP server. This parameter is mandatory.	**/opt/omm/oms/auditLog** (example value)
SFTP Username	Specifies the username for logging in to the SFTP server. This parameter is mandatory.	**root** (example value)
SFTP Password	Specifies the password for logging in to the SFTP server. This parameter is mandatory.	*Password for logging in to the SFTP server*
SFTP Public key	Specifies the public key of the SFTP server. This parameter is optional. You are advised to set the public key of the SFTP server. Otherwise, security risks may exist.	-
Dumping Mode	Specifies the dump mode. This parameter is mandatory. • **By Quantity**: If the number of pieces of logs reaches the value of this parameter (**100000** by default), the logs are dumped. • **By Time**: specifies the date when logs are dumped. The dumping frequency is once a year.	• By Quantity • By Time
Dumping Date	This parameter is mandatory. It is available when the dump mode is set to **By Time**. After you select a dump date, the system starts dumping on this date. The logs to be dumped include all the audit logs generated before January 1 00:00 of the current year.	November 06 (example)

NOTE

If the SFTP public key is empty, the system prompts a security risk message. Determine the security risk, then save the configuration.

Step 5 Click **OK** to complete the settings.

NOTE

Key fields in the audit log dump file are as follows:

- **USERTYPE** indicates the user type. Value **0** indicates the Human-machine user, and value **1** indicates the Machine-machine user.
- **LOGLEVEL** indicates the security level. Value **0** indicates Critical, value **1** indicates Major, value **2** indicates Minor, and value **3** indicates Warning.
- **OPERATERESULT** indicates the operation result. Value **0** indicates that the operation is successful, and value **1** indicates that the operation is failed.

----End

10.7 Tenant Resources

10.7.1 Introduction to Multi-Tenant

10.7.1.1 Overview

Definition

Multi-tenant specifies multiple resource sets (a resource set is a tenant) in a MRS big data cluster and is able to allocate and schedule resources. The resources include computing resources and storage resources.

Background

Modern enterprise data clusters are developing towards centralization and cloudification. Enterprise-class big data clusters must meet the following requirements:

- Carry data of different types and formats and run jobs and applications of different types (analysis, query, and stream processing).
- Isolate data of a user from that of another user who has demanding requirements on data security, such as a bank or government institute.

The preceding requirements bring the following challenges to the big data cluster:

- Proper allocation and scheduling of resources to ensure stable operating of applications and jobs
- Strict access control to ensure data and service security

Multi-tenant isolates the resources of a big data cluster into resource sets. Users can lease desired resource sets to run applications and jobs and store data. In a big data cluster, multiple resource sets can be deployed to meet diverse requirements of multiple users.

The MRS enterprise-class big data cluster provides a complete enterprise-class big data multi-tenant solution.

Highlights

- Proper resource configuration and isolation

  The resources of a tenant are isolated from those of another tenant. The resource use of a tenant does not affect other tenants. This mechanism ensures that each tenant can configure resources based on service requirements, improving resource utilization.
- Resource consumption measurement and statistics

  Tenants are system resource applicants and consumers. System resources are planned and allocated based on tenants. Resource consumption by tenants can be measured and recorded.
- Ensured data security and access security

  In multi-tenant scenarios, the data of each tenant is stored separately to ensure data security. The access to tenants' resources is controlled to ensure access security.

10.7.1.2 Technical Principles

10.7.1.2.1 Multi-Tenant Management

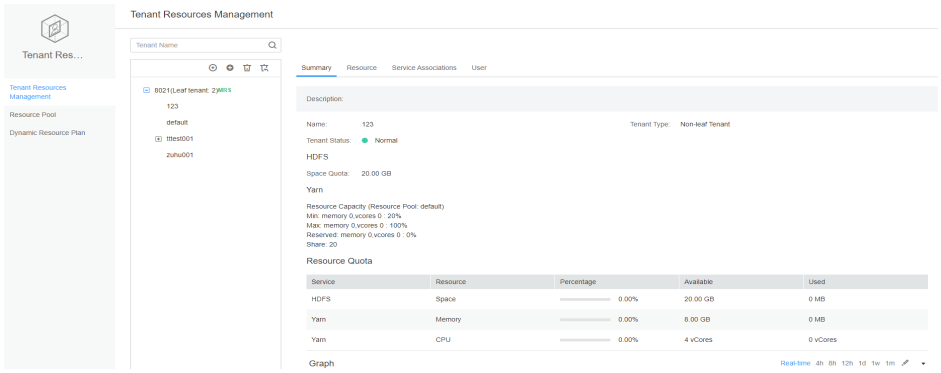
Unified Multi-Tenant Management

Log in to FusionInsight Manager and choose **Tenant Resources** > **Tenant Resources Management**. On the page that is displayed, you can find that FusionInsight Manager is a unified multi-tenant management platform that integrates multiple functions such as tenant lifecycle management, tenant resource configuration, tenant service association, and tenant resource usage statistics, delivering a mature multi-tenant management model and achieving centralized tenant and service management.

Graphical User Interface

FusionInsight Manager provides the graphical multi-tenant management interface and manages and operates multiple levels of tenants using the tree structure. Additionally, FusionInsight Manager integrates the basic information and resource quota of the current tenant in one interface to facilitate O&M and management, as shown in **Figure 10-14**.

Figure 10-14 Tenant management page of FusionInsight Manager



Level-based Tenant Management

FusionInsight Manager supports a level-based tenant management model in which you can add sub-tenants to an existing tenant to re-configure resources. Sub-tenants of level-1 tenants belong to level-2 tenants, and so on. FusionInsight Manager provides enterprises with a field-tested multi-tenant management model, enabling centralized tenant and service management.

Simplified Rights Management

In FusionInsight Manager, common users are shielded from internal rights management details and administrators' rights management operations are simplified, improving rights management usability and user experience.

- FusionInsight Manager adopts the role-based access control (RBAC) mode to configure rights for users as required during multi-tenant management.
- Administrator of tenants, the administrator has tenants' management rights, including viewing resources and services of the current tenant, adding or deleting sub-tenants of the current tenant, and managing rights of sub-tenants' resources. The administrator of a single tenant can be defined and the management over a tenant can be delegated to another user except the system administrator.
- Roles corresponding to tenants, roles have all rights on the computing resources and storage resources of a tenant. During the creation of a tenant, the system automatically creates a corresponding role. You can add a user and bind the user to the tenant role so that it can use the resources of the tenant.

Clear Resource Management

- **Self-Service Resource Configuration**

  In FusionInsight Manager, you can configure the computing resources and storage resources during the creation of a tenant and add, modify, or delete the resources of a tenant.

  Rights of the role that corresponds to the current tenant are updated automatically when you modify the computing resources and storage resources of a tenant.

- **Resource Usage Statistics**

  Resource usage statistics is critical for administrators to make O&M decisions based on the status of cluster applications and services, improving the cluster O&M efficiency. The FusionInsight Manager displays the resource statistics of tenant through the **Resource Quota**, including the dynamic computing resource VCores and Memory of tenant and the usage statistics of HDFS storage resources (Space).

NOTE

- **Resource Quotas** dynamically calculates the resource usage of tenants.

Service	Resource	Percentage	Available	Used
HDFS	Space	0.00%	20.00 GB	0 MB
Yarn	Memory	0.00%	8.00 GB	0 MB
Yarn	CPU	0.00%	4 vCores	0 vCores

  The available resources of the Superior scheduler are calculated as follows:

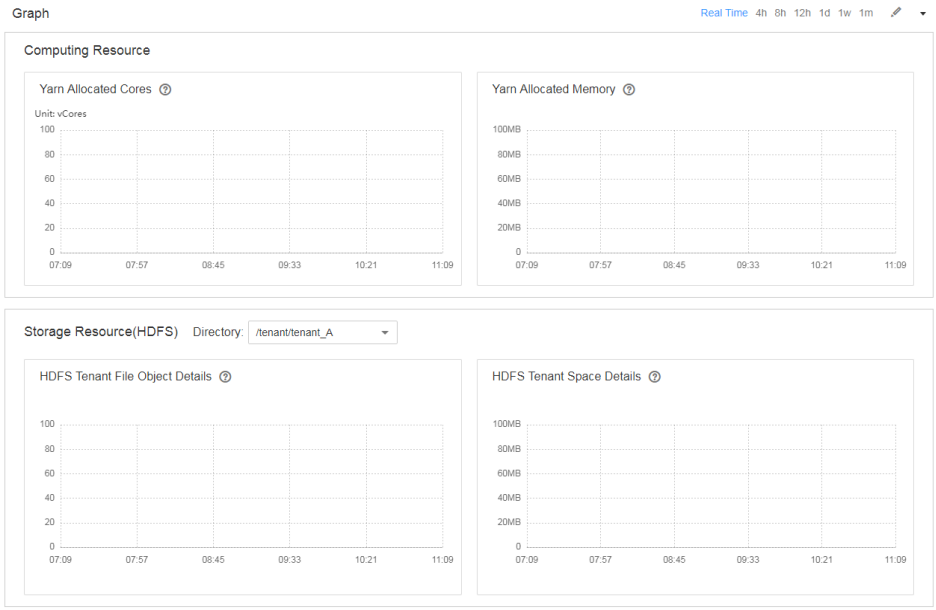
  - Superior

    The available Yarn resources (memory and CPU) are allocated in proportion based on the queue weight.
- When the tenant administrator is bound to a tenant role, the tenant administrator has the rights to manage the tenant and use all resources of the tenant.

- **Graphical Resource Monitoring**

  The resource graphical monitoring supports the graphical display of monitoring items listed in **Table 10-29**, as shown in **Figure 10-15**.

Figure 10-15 Refined monitoring



By default, the real-time monitoring data is displayed. You can click to customize a time range. The default time ranges include 4 hours, 8 hours, 12 hours, 1 day, 1 week, and 1 month. Click and choose **Export** from the shortcut menu to export the monitoring item information.

Table 10-29 Item

Service	Metric	Description
HDFS	HDFS Tenant Space Details • Allocated Space • Used Space	HDFS can select a specified storage directory for monitoring. The storage directory is the same as the directory added by the current tenant in **Resource**.
	HDFS Tenant File Object Details • Number of Used File Objects	
Yarn	Yarn Allocated Cores • Maximum Number of CPU Cores in an AM • Allocated Cores • Number of Used CPU Cores in an AM	Monitoring information of the current tenant can be displayed. If no subitem is configured for a tenant, this information is not displayed. The monitoring data is obtained from **Scheduler** > **Application Queues** > **Queue:** *tenant name* on the native WebUI of Yarn.
	Yarn Allocated Memory • Allocated Maximum AM Memory • Allocated Memory • Used AM Memory	

10.7.1.2.2 Models Related to Multi-Tenant

Models Related to Multi-Tenant

Figure 10-16 shows the models related to multi-tenant.

Figure 10-16 Models related to multi-tenant

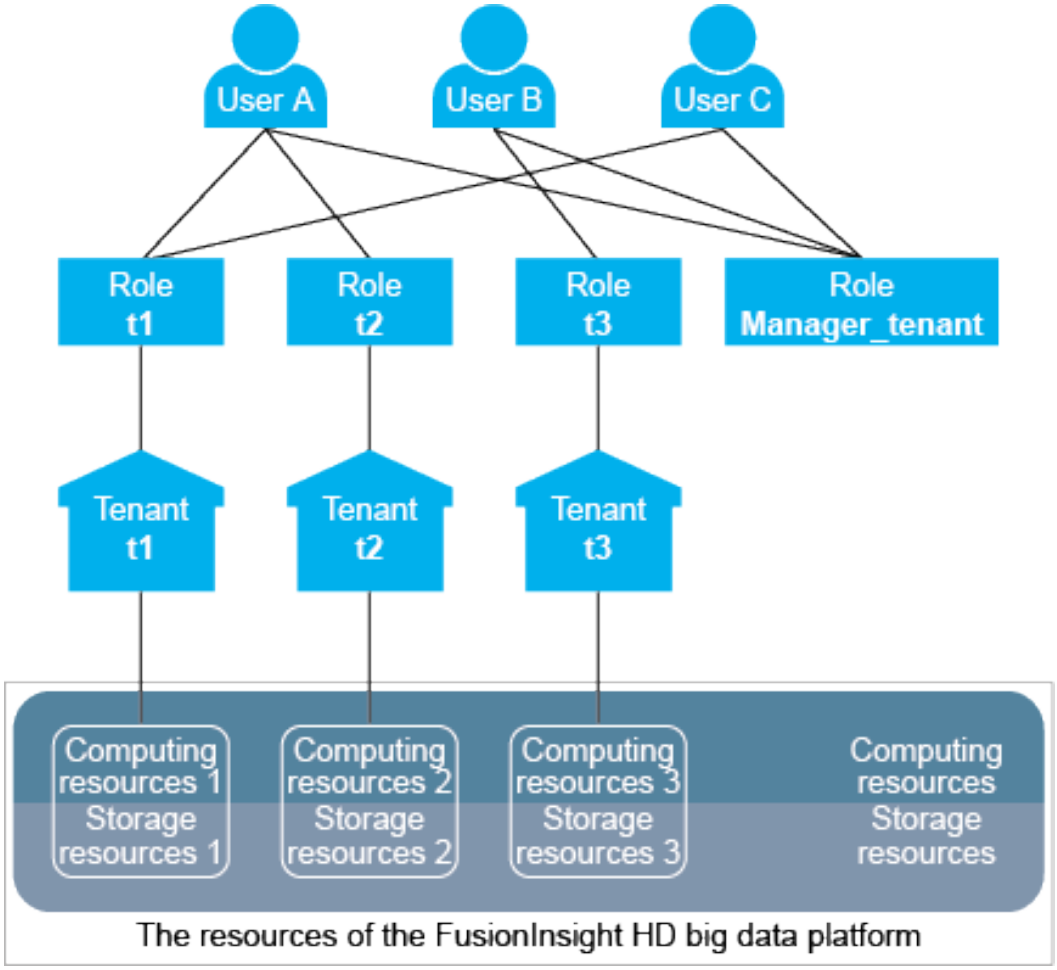


Table 10-30 describes the concepts involved in **Figure 10-16**.

Table 10-30 Concepts involved

Concept	Description
User	A natural person who has a name and password and uses the big data platform. **Figure 10-16** shows three different users: user A, user B, and user C.

Concept	Description
Role	A role is a carrier of one or more rights. Rights are assigned to specific objects, for example, access rights for the **/tenant** directory in HDFS. **Figure 10-16** shows four roles: role **t1**, role **t2**, role **t3**, and role **Manager_tenant**. • Roles **t1**, **t2**, and **t3** are automatically generated when tenants are created. The role names are the same as the tenant names. That is, roles **t1**, **t2**, and **t3** map to tenants **t1**, **t2**, and **t3**. Role names and tenant names need to be used together. • Role **Manager_tenant** is the role of the cluster and cannot be used separately.
Tenant	A tenant is a resource set divided from a big data cluster. Multi-tenant refers to multiple tenants. The source sets further divided in a tenant are called sub-tenants. **Figure 10-16** shows three tenants: tenant **t1**, tenant **t2**, and tenant **t3**.
Resource	• Computing resources include CPUs and memory. The computing resources of a tenant are divided from the total computing resources of the cluster. One tenant cannot occupy the computing resources of another tenant. In **Figure 10-16**, computing resources 1, 2, and 3 are divided from the cluster's computing resources by tenants **t1**, **t2**, and **t3**. • Storage resources include disks and third-party storage systems. The storage resources of a tenant are divided from the total storage resources of the cluster. One tenant cannot occupy the storage resources of another tenant. In **Figure 10-16**, storage resources 1, 2, and 3 are divided from the cluster's storage resources by tenants **t1**, **t2**, and **t3**.

If a user wants to use a tenant's resources or add or delete a sub-tenant from a tenant, the user needs to be bound to both the tenant role and role **Manager_tenant**. **Table 10-31** shows the roles bound to each user in **Table 10-31**.

Table 10-31 Roles bound to each user

User	Role	Rights
User A	• Role **t1** • Role **t2** • Role **Manager_tenant**	• Uses the resources of tenants **t1** and **t2**. • Adds or deletes sub-tenants for tenants **t1** and **t2**.
User B	• Role **t3** • Role **Manager_tenant**	• Uses the resources of tenant **t3**. • Adds or deletes sub-tenants for tenant **t3**.
User C	• Role **t1** • Role **Manager_tenant**	• Uses the resources of tenant **t1**. • Adds or deletes sub-tenants for tenant **t1**.

One user can be bound to multiple roles, and one role can be bound to multiple users. Users are associated with tenants by binding themselves to the tenants. For this reason, tenants and users are in many-to-many relationship. One user can use the resources of multiple tenants, and multiple users can use the resources of a tenant. In **Figure 10-16**, user A uses the resources of tenants **t1** and **t2**, and users A and C uses the resources of tenant **t1**.

NOTE

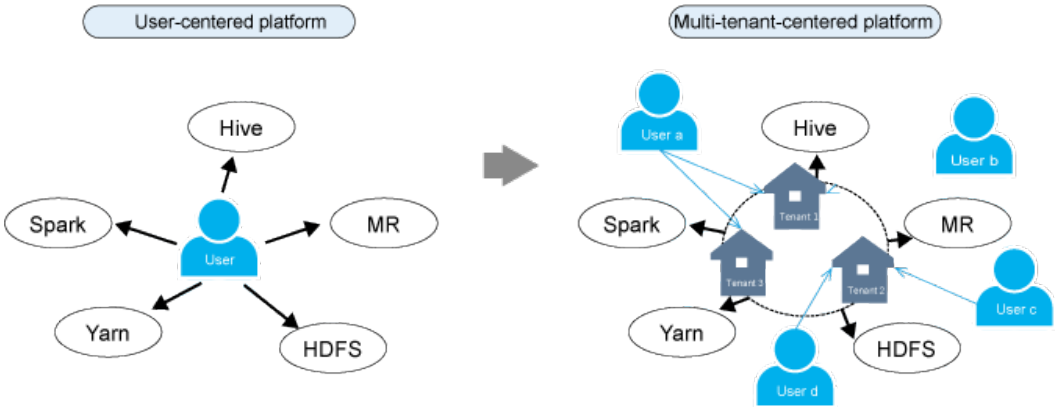
The parent tenant, sub-tenant, level-1 tenants, and level-2 tenants are designed for multi-tenant service scenarios. Pay attention to the differences between these concepts and those of leaf tenant and non-leaf tenant on FusionInsight Manager.

- Level-1 tenant: The name is determined by the tenant's level. For example, the created tenant is a level-1 tenant whose sub-tenant is a level-2 tenant.
- Parent tenant and sub-tenant: indicates the hierarchical relationship between tenants.
- Non-leaf tenant resource: indicates the tenant resource type selected during tenant creation, which can be used to create sub-tenants.
- Leaf tenant resource: indicates the tenant resource type selected during tenant creation, which cannot be used to create sub-tenants.

Multi-Tenant Platform

Tenant is a core concept of the FusionInsight big data platform. It assists in transforming the big data platform from the user-centered platform to the multi-tenant-centered platform to better cope with the multi-tenant application environment of modern enterprises.

Figure 10-17 User-centered platform and multi-tenant-centered platform



On the user-centered big data platform, users can directly access and use all resources and services.

- However, some cluster resources may not be used, lowering resource utilization.
- The data of different users may be stored together, decreasing data security.

On the multi-tenant-centered big data platform, users use required resources and services by accessing the tenants.

- Resources are allocated and scheduled based on application requirements and used based on tenants, increasing resource utilization.
- Users can access the resources of tenants only after being assigned roles, enhancing access security.
- The data of tenants is isolated, ensuring data security.

10.7.1.2.3 Resource Overview

The resources of the MRS big data platform are divided into computing resources and storage resources. Multi-tenant enables resource isolation:

- **Computing Resource**

  Computing resources include CPUs and memory. One tenant cannot occupy the computing resources of another tenant.
- **Storage Resource**

  Storage resources include disks and third-party storage systems. One tenant cannot access the data of another tenant.

Computing Resource

Computing resources are divided into static service resources and dynamic resources.

- **Static service resources**

  Static service resources are computing resources allocated to each service. The total volume of computing resources allocated to each service is fixed. Such services include Flume, HBase, HDFS and Yarn.
- **Dynamic resources**

Dynamic resources are computing resources dynamically scheduled to a task queue by distributed resource management service Yarn. Yarn dynamically schedules resources for the task queues of Mapreduce, Spark2x, Flink, and Hive.

NOTE

The resources allocated to Yarn in a big data cluster are static service resources and can be dynamically allocated to task queues by Yarn.

Storage Resource

Storage resources are data storage resources that can be allocated by distributed file storage service HDFS. Directories are used as the basic unit of HDFS storage resource allocation. Tenants can obtain storage resources by specifying directories in the HDFS file system.

10.7.1.2.4 Dynamic Resources

Overview

Yarn provides the distributed resource management function for a big data cluster. The total volume of resources allocated to Yarn can be configured. Then Yarn allocates and schedules computing resources for task queues. The computing resources of Mapreduce, Spark, Flink and Hive task queues are allocated and scheduled by Yarn.

Yarn queues are basic units of computing resource allocation.

For tenants, the resources obtained using Yarn task queues are dynamic resources. Users can dynamically create and modify the quotas of task queues and view the status and statistics of task queues.

Resource Pool

Complex cluster environments and upper-layer requirements are facing enterprise IT systems. For example:

- Heterogeneous cluster: The computing speed, storage capacity, and network performance of each node in the cluster are different. All the tasks of complex applications need to be properly allocated to each compute node in the cluster based on service requirements.
- Computing isolation: Data must be shared among multiple departments but computing resources must be distributed onto different compute nodes.

Compute nodes must be partitioned.

Resource pools are used to specify the configuration of dynamic resources. Yarn task queues are associated with resource pools for resource allocation and scheduling.

Only one default resource pool can be set for a tenant. Users can bind to the role of a tenant to use the resources in the resource pool of the tenant. If resources in multiple resource pools need to be used, users can bind themselves to multiple tenant roles.

Scheduling Mechanism

Yarn dynamic resources support label based scheduling. This policy creates labels for compute nodes (Yarn NodeManager nodes) of Yarn clusters and adds the compute nodes with the same label into the same resource pool. Then Yarn dynamically associates the task queues with resource pools based on the resource requirements of the task queues.

For example, a cluster has more than 40 nodes. Labels Normal, HighCPU, HighMEM, and HighIO are created based on the hardware and network configurations of nodes and added four resource pools. **Table 10-32** describes the performance of each node in the resource pool.

Table 10-32 Performance of each node in a resource pool

Label	Number of Nodes	Hardware and Network Configuration	Added To	Association
Normal	10	Minor	Resource pool A	Common task queue
HighCPU	10	High-performance CPU	Resource pool B	Computing-intensive task queue
HighMEM	10	Large memory	Resource pool C	Memory-intensive task queue
HighIO	10	High-performance network	Resource pool D	I/O-intensive task queue

Task queues can use the compute nodes in the associated resource pools only.

- Common task queues are associated with resource pool A and use nodes with hardware and network configurations labeled with Normal.
- Computing-intensive task queues are associated with resource pool B and use nodes with CPUs labeled with HighCPU.
- Memory-intensive task queues are associated with resource pool C and use nodes with memory labeled with HighMEM.
- I/O-intensive task queues are associated with resource pool C and use nodes with the network labeled with HighIO.

Yarn task queues are associated with specified resource pools to efficiently utilize resources in resource pools and ensure node performance.

FusionInsight Manager supports a maximum of add 50 resource pools. A default resource pool is included in the system by default.

Introduction to Schedulers

By default, the Superior scheduler is enabled for the MRS cluster.

- The Superior scheduler is an enhanced version and named after the Lake Superior, indicating that the scheduler can manage a large amount of data.

To meet enterprise requirements and tackle challenges facing the Yarn community in scheduling. The Superior scheduler provides the following enhancements:

- Enhanced resource sharing policy

  The Superior scheduler supports queue hierarchy. It integrates the functions of open source schedulers and shares resources based on configurable policies. In terms of instances, administrators can use the Superior scheduler to configure an absolute value or a percentage policy for queue resources. The resource sharing policy of the Superior scheduler enhances the label scheduling policy of Yarn as a resource pool feature. Nodes in a Yarn cluster can be grouped based on the capacity or service type to ensure that queues can more efficiently utilize resources.

- Tenant-based resource reservation policy

  Resources required by tenants must be ensured for running critical tasks. The Superior scheduler builds a resource reservation mechanism. With this mechanism, reserved resources can be allocated to tasks run by tenant queues in a timely manner to ensure proper task execution.

- Fair sharing among tenants and resource pool users

  The Superior scheduler allows shared resources to be configured for users in a queue. Each tenant may have users with different weights. Heavily weighted users may require more shared resources.

- Ensured scheduling performance in a big cluster

  The Superior scheduler receives heartbeats from each NodeManager and saves resource information in memory, which enables the scheduler to control cluster resource usage globally. The Superior scheduler uses the push scheduling model, which makes the scheduling more precise and efficient and remarkably improves cluster resource utilization. Additionally, the Superior scheduler delivers excellent performance when the interval between NodeManager heartbeats is long and prevents heartbeat storms in big clusters.

- Priority policy

  If the minimum resource requirement of a service cannot be met after the service obtains all available resources, a preemption occurs. The preemption function is disabled by default.

10.7.1.2.5 Storage Resource

Overview

As a distributed file storage service in a big data cluster, HDFS stores all the user data of the upper-layer applications in the big data cluster, including the data written to HBase tables or Hive tables.

Directories are used as the basic unit of HDFS storage resource allocation. HDFS supports the conventional hierarchical file structure. Users can create directories and create, delete, move, or rename files in directories. Tenants can obtain storage resources by specifying directories in the HDFS file system.

Scheduling Mechanism

HDFS directories can be stored on nodes with specified labels or disks of specified hardware types. For example:

- When both real-time query and data analysis tasks are running in one cluster, the real-time query tasks are deployed on some nodes; therefore, the queried data must be stored on these nodes.
- Based on actual service requirements, key data needs to be stored on nodes with high reliability.

Administrators can flexibly configure HDFS data storage policies based on actual service requirements and data features to store data on specified nodes.

For tenants, storage resources indicate the HDFS resources occupied by them. They can implement storage resource scheduling by storing data of specified directories in storage paths configured by tenants to ensure data isolation between tenants.

Users can add or delete HDFS storage directories of tenants and set the file quantity quota and storage capacity quota of directories to manage storage resources.

10.7.1.3 Multi-Tenant Use

10.7.1.3.1 Overview

Tenants are used in resource control and service isolation scenarios. Administrators need to confirm the service scenarios of cluster resources, and then plan tenants.

NOTE

- By default, the Yarn component of the newly installed cluster uses the Superior scheduler. For details, see **Using the Superior Scheduler in Multi-Tenant Scenarios**.

Multi-tenant involves three types of operations: creating a tenant, managing tenants, and managing resources. **Table 10-33** describes the operations.

Table 10-33 Operations involved in multi-tenant

Operation	Action	Description
Creating a tenant	• Adding a tenant • Adding a sub-tenant • Creating a user and binding the user to the role of a tenant	During the creation of a tenant, you can configure its computing resources, storage resources, and associated services based on service requirements. In addition, you can add users to the tenant and bind necessary roles to these users. A user who creates level-1 tenants must be bound to the **Manager_administrator** or **System_administrator** role. A user who creates sub-tenants must be bound to the role of the parent tenant at least.
Managing tenants	• Managing tenant directories • Restoring tenant data • Clearing unassociated queues of a tenant • Deleting a tenant	Modifies tenants as the services change. A user who manages or deletes level-1 tenants and restores tenants' data must be bound to the **Manager_administrator** or **System_administrator** role. A user who manages or deletes sub-tenants must be bound to the role of the parent tenant at least.
Managing resources	• Adding a resource pool • Modifying a resource pool • Deleting a resource pool • Configuring a queue • Configuring the queue capacity policy of a resource pool • Clearing queue configurations	Reconfigure resources for tenants as the services change. A user who manages resources must be bound to the **Manager_administrator** or **System_administrator** role.

10.7.1.3.2 Process Overview

Administrators need to confirm the service scenarios of cluster and plan user rights. After that, administrators need to add tenants and configure dynamic resources, storage resources, and related services for tenants on FusionInsight Manager.

Figure 10-18 shows the procedure of creating a tenant.

Figure 10-18 Creating a tenant

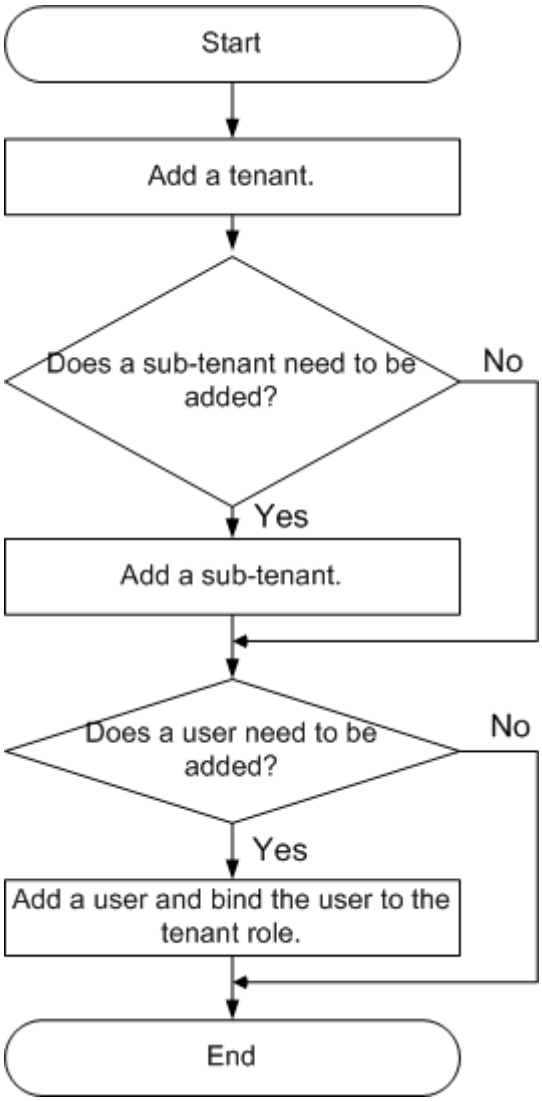


Table 10-34 describes the operations.

Table 10-34 Description on creating a tenant

Operation	Description
Adding a Tenant	Configures computing resources, storage resources, and related services for tenants.
Adding a Sub-Tenant	Configures computing resources, storage resources, and related services for tenants.
Adding a User and Binding the User to a Tenant Role	If a user wants to use the resources of **tenant1** or add or delete a sub-tenant for **tenant1**, the user must be bound with the **Manager_tenant** and **tenant1**_*cluster ID* roles.

10.7.2 Using the Superior Scheduler in Multi-Tenant Scenarios

10.7.2.1 Creating Tenants

10.7.2.1.1 Adding a Tenant

Scenario

Based on the resource consumption and isolation plan and requirements of services, administrators can create tenants on FusionInsight Manager to meet actual application scenarios.

Prerequisites

- A tenant name has been planned based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
- Resources to be allocated to the current tenant have been planned to ensure that the sum of capacities of direct sub-tenants at every level cannot exceed the current tenant.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 Click ⊕. On the displayed page, configure tenant properties based on **Table 10-35**.

Table 10-35 Tenant parameters

Parameter	Description
Cluster	Select the cluster for which you want to create a tenant.
Name	• Specifies the name of the current tenant. The value consists of 3 to 50 characters, which can be letters, digits, or underscores (_). • Plan a tenant name based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
Tenant Type	Specifies whether the specified tenant is a leaf tenant. • When **Leaf Tenant** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. • When **Non-leaf Tenant** is selected, the current tenant is not a leaf tenant and sub-tenants can be added to the current tenant.

Parameter	Description
Computing Resource	Specifies the dynamic computing resources for the current tenant. ● When **Yarn** is selected, the system automatically creates a task queue in Yarn and the queue is named the same as the name of the tenant. – A leaf tenant can directly submit tasks to the task queue. – A non-leaf tenant cannot directly submit tasks to the task queue. However, Yarn adds an extra task queue (hidden) named **Default** for the non-leaf tenant to record the remaining resource capacity of the tenant. Actual tasks do not run in this queue. ● When dynamic resources are not **Yarn** resources, the system does not automatically create a task queue.
Configuration Mode	Indicates the configuration mode of computing resource parameters. ● If you select **Basic**, you only need to set **Default Resource Pool Capacity (%)**. ● If you select **Advanced**, you can manually configure the resource allocation weight and the minimum, maximum, and reserved resources of a tenant.
Default Resource Pool Capacity (%)	Specifies the computing resource usage in the default resource pool of the current tenant. The value ranges from 0 to 100%.
Weight	Resource allocation weight. The value ranges from 0 to 100.
Minimum Resource	Resources guaranteed for the tenant resource (preemption supported). The value can be a percentage of the parent tenant resource's resources or an absolute value. When a tenant resource has a light workload, the resources of the tenant resource are automatically allocated to other tenant resources. When the available resources of the tenant resource do not meet the minimum threshold, the tenant resource can preempt the resources lent to other tenant resources.
Maximum Resource	Maximum resources that a tenant resource can use. The value can be a percentage of the parent tenant resource's resources or an absolute value.

Parameter	Description
Reserved Resource	Resources reserved for a tenant resource. Even when a tenant resource has no workload, other tenant resources cannot use the reserved resources of the tenant resource. The value can be a percentage of the parent tenant resource's resources or an absolute value.
Storage Resource	Specifies storage resources of the current tenant. ● When **HDFS** is selected, the system automatically allocates storage resources. ● When **HDFS** is not selected, the system does not automatically allocate storage resources.
Quota	Specifies the file and directory quantity quota.
Space Quota	Specifies the used HDFS storage space quota of the current tenant. ● Value range: When Space Quota unit is set to **MB**, this parameter ranges from **1** to **8796093022208**. When Space Quota unit is set to **GB**, this parameter ranges from **1** to **8589934592**. ● This parameter indicates the maximum HDFS storage space that can be used by the tenant, but does not indicate the actual space used. ● If the value is greater than the size of the HDFS physical disk space, the maximum space that can be used is all the HDFS physical disk space.
Storage Path	Specifies the HDFS storage directory for a tenant. ● The system creates a file folder named after the tenant name in the **/tenant** directory by default. For example, the default HDFS storage directory for **ta1** is **/tenant/ta1**. ● When a tenant is created for the first time, the system creates the **/tenant** directory in the HDFS root directory. The storage path is customizable.
Service	For details about whether to associate resources of other services, see **Step 4**.
Description	Configure the description of the current tenant.

NOTE

During the creation of a tenant, the system automatically creates a corresponding role, the computing resources, and the storage resources.

- The new role has the rights on the computing resources and storage resources. The role and its rights are controlled by the system automatically and cannot be controlled manually under **System > Permission > Role**. The role name is *tenant name_cluster ID*. By default, the cluster ID of the first cluster is not displayed.
- When using this tenant, create a system user and bind the user to the role of the tenant. For details, see **Adding a User and Binding the User to a Tenant Role**.
- During the creation of a tenant, the system automatically creates a Yarn task queue named after the tenant. If the queue name exists, the new queue is named **Tenant name-N**. **N** indicates a natural number starting from 1. When a same name exists, the value **N** increases automatically to differentiate the queue from others. For example, **saletenant**, **saletenant-1**, and **saletenant-2**.

Step 3 Whether the current tenant need to associate resources of other services.

- If yes, go to **4**.
- If no, go to **Step 5**.

Step 4 Click **Associated Service** to configure other service resources used by the current tenant.

1. Select **HBase** in **Service**.
2. Make a selection in **Association Type**:
   - **Exclusive** indicates service resources used by the tenant exclusively. Other tenants cannot associate with this service.
   - **Share** indicates shared service resources, which can be used by other tenants.

NOTE

- When creating a tenant, you can only associate HBase with the tenant. For existing tenants, you can associate the following services: HDFS, HBase, and Yarn.
- Associating existing tenants with service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page and click **Associated Service** to associate the current tenant with service services.
- Canceling the association between existing tenants and service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page, click **Delete**, select **I have read the information and understand the impact.**, and click **OK** to cancel the association with service.

3. Click **OK**.

Step 5 Click **OK**. When **Tenant created successfully.** is displayed on the page, the tenant is added successfully.

----End

10.7.2.1.2 Adding a Sub-Tenant

Scenario

Based on the resource consumption and isolation plan and requirements of services, administrators can create Sub-Tenants on FusionInsight Manager, and allocate resources of the current tenant to meet the actual application scenario.

Prerequisites

- A parent tenant has been added, and belongs to a non-leaf tenant.
- A tenant name has been planned based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
- Resources to be allocated to the current tenant have been planned to ensure that the sum of capacities of direct sub-tenants at every level cannot exceed the current tenant.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, move the cursor to the tenant node to which the sub-tenant is added. Click  . In the displayed window, configure the sub-tenant properties based on **Table 10-36**.

Table 10-36 Sub-tenant parameters

Parameter	Description
Cluster	Specifies the cluster of the parent tenant.
Parent Tenant Resource	Specifies the name of the parent tenant.
Name	• Specifies the name of the current tenant. The value consists of 3 to 50 characters, which can be letters, digits, or underscores (_). • Plan a sub-tenant name based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
Tenant Type	Specifies whether the specified tenant is a leaf tenant. • When **Leaf Tenant** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. • When **Non-leaf Tenant** is selected, the current tenant is not a leaf tenant and sub-tenants can be added to the current tenant. However, the tenant depth cannot exceed 5 levels.

Parameter	Description
Computing Resource	Specifies the dynamic computing resources for the current tenant. ● When **Yarn** is selected, the system automatically creates a task queue in Yarn and the queue is named the same as the name of the tenant. – A leaf tenant can directly submit tasks to the task queue. – A non-leaf tenant cannot directly submit tasks to the task queue. However, Yarn adds an extra task queue (hidden) named **Default** for the non-leaf tenant to record the remaining resource capacity of the tenant. Actual tasks do not run in this queue. ● When dynamic resources are not **Yarn** resources, the system does not automatically create a task queue.
Configuration Mode	Indicates the configuration mode of computing resource parameters. ● If you select **Basic**, you only need to set **Default Resource Pool Capacity (%)**. ● If you select **Advanced**, you can manually configure the resource allocation weight and the minimum, maximum, and reserved resources of a tenant.
Default Resource Pool Capacity (%)	Specifies the computing resource usage of the current tenant. The base value is the total resources of the parent tenant.
Weight	Resource allocation weight. The value ranges from 0 to 100.
Minimum Resource	Resources guaranteed for the tenant resource (preemption supported). The value can be a percentage of the parent tenant resource's resources or an absolute value. When a tenant resource has a light workload, the resources of the tenant resource are automatically allocated to other tenant resources. When the available resources of the tenant resource do not meet the minimum threshold, the tenant resource can preempt the resources lent to other tenant resources.
Maximum Resource	Maximum resources that a tenant resource can use. The value can be a percentage of the parent tenant resource's resources or an absolute value.

Parameter	Description
Reserved Resource	Resources reserved for a tenant resource. Even when a tenant resource has no workload, other tenant resources cannot use the reserved resources of the tenant resource. The value can be a percentage of the parent tenant resource's resources or an absolute value.
Storage Resource	Specifies storage resources of the current tenant. • When **HDFS** is selected, the system automatically creates a file in the HDFS parent tenant directory. The file is named the same as the name of the sub-tenant. • When **HDFS** is not selected, the system does not automatically allocate storage resources.
Quota	Specifies the file and directory quantity quota.
Space Quota	Specifies the used HDFS storage space quota of the current tenant. • When **Space Quota Unit** is set to **MB**, this parameter ranges from **1** to **8796093022208**. When **Space Quota Unit** is set to GB, this parameter ranges from **1** to **8589934592**. The maximum value of this parameter does not exceed the total storage quota of the parent tenant. • This parameter indicates the maximum HDFS storage space that can be used by the tenant, but does not indicate the actual space used. • If the value is greater than the size of the HDFS physical disk space, the maximum space that can be used is all the HDFS physical disk space. • If this quota is greater than the quota of the parent tenant, the actual storage space will be affected by the quota of the parent tenant.
Storage Path	Specifies the HDFS storage directory for a tenant. • The system creates a file folder named after the sub-tenant name in the directory of the parent tenant by default. For example, if the sub-tenant is **ta1s** and the parent directory is **/tenant/ta1**, the system sets the **Storage Path** for the sub-tenant to **/tenant/ta1/ta1s**. • The storage path is customizable in the parent directory.
Service	For details about whether to associate resources of other services, see **Step 4**.
Description	Configure the description of the current tenant.

NOTE

During the creation of a tenant, the system automatically creates a corresponding role, the computing resources, and the storage resources.

- The new role has the rights on the computing resources and storage resources. The role and its rights are controlled by the system automatically and cannot be controlled manually under **System > Permission > Role**. The role name is *tenant name_cluster ID*. By default, the cluster ID of the first cluster is not displayed.
- When using this tenant, create a system user and bind the user to the role of the tenant. For details, see **Adding a User and Binding the User to a Tenant Role**.
- The sub-tenant can further allocate the resources of the current tenant. The sum of the resource percentage of direct sub-tenants of a parent tenant cannot exceed 100%. The sum of the computing resource percentage of all level-1 tenants cannot exceed 100%.

Step 3 Whether the current tenant need to associate resources of other services.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Click **Associated Service** to configure other service resources used by the current tenant.

1. Select **HBase** in **Service**.
2. Make a selection in **Association Type**:
   - **Exclusive** indicates service resources used by the tenant exclusively. Other tenants cannot associate with this service.
   - **Share** indicates shared service resources, which can be used by other tenants.

NOTE

- When creating a tenant, you can only associate HBase with the tenant. For existing tenants, you can associate the following services: HDFS, HBase, and Yarn.
- Associating existing tenants with service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page and click **Associated Service** to associate the current tenant with service services.
- Canceling the association between existing tenants and service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page, click **Delete**, select **I have read the information and understand the impact.**, and click **OK** to cancel the association with service.

3. Click **OK**.

Step 5 Click **OK**. When **Tenant created successfully.** is displayed on the page, the tenant is added successfully.

----End

10.7.2.1.3 Adding a User and Binding the User to a Tenant Role

Scenario

The created tenant cannot directly log in to the cluster to access resources. Administrators need to create a user for a tenant on FusionInsight Manager and bind the user to a tenant role to assign operation rights to the user.

Prerequisites

The system administrator has understood service requirements and created a tenant.

Procedure

Step 1 On FusionInsight Manager, click **System > Permission > User**.

Step 2 To add a user to the system, click **Create**.

To bind tenant rights to an existing user in the system, click **Modify** in the column where the user locates. The configuration page is displayed.

For details about configuring parameters of a user, see **Table 10-37**.

Table 10-37 User parameters

Parameter	Description
Username	Specifies the name of the current tenant. The value consists of 3 to 32 characters, which can be letters, digits, underlines (_), hyphens(-), or spaces. • **Username** cannot be the same as any username of the OS on each node in the cluster. Otherwise, the user account cannot be used properly. • Usernames of the same letters but different cases are not supported. For example, if **User1** already exists, user **user1** cannot be created. When using user **User1**, enter the correct username.
User Type	Options include **Human-Machine** and **Machine-Machine**. • **Human-Machine** user: Used in FusionInsight Manager O&M scenarios and component client operation scenarios. If you select **Human-Machine**, you need to set **Password** and **Confirm password**. • **Machine-Machine** user: Used in application development scenarios. If you select **Machine-Machine**, the user password is generated randomly.
Password	If you select **Human-Machine**, set **Password**. The password must contain 8 to 64 characters, consisting at least 4 of uppercase letters, lowercase letters, digits, and special characters and spaces. Cannot be the username or username spelled backwards.
Confirm Password	Enter the password again.

Parameter	Description
User Group	In **User Group**, click **Add** to add the user to a user group. ● If a role is added to a user group, users in the user group can obtain the rights of the role. ● For example, assign Hive rights to the new user and add the user to the **Hive** group.
Primary Group	Select a group as the primary group of directories and files of the user. The drop-down list contains groups that are selected in **User Group**.
Role	Click **Add** to add a role to the user as required. **NOTE** ● If a user wants to use resources allocated to **tenant1** add sub-tenants to or delete sub-tenants from **tenant1**, bind the **Manager_tenant** and **tenant1**_*cluster ID* roles to the user. ● If the tenant is associated with the HBase service and Ranger authentication is enabled for the current cluster, you need to configure the HBase execution permission on the Ranger WebUI.
Description	Configure the description of the current user.

Step 3 Click **OK**.

----End

10.7.2.2 Managing Tenants

10.7.2.2.1 Managing a Tenant Directory

Scenario

The administrator manages the HDFS storage directory used by a specified tenant on FusionInsight Manager based on service requirements. The management operations include adding tenant directories, modifying quantity quotas of files and directories, and storage space quota of the directory, and deleting directories.

Prerequisites

Tenants with HDFS storage resources are added.

Procedure

View a tenant directory.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click a target tenant.

Step 3 Click **Resource**.

Step 4 View the **HDFS Storage** table.

- The **Quota** column indicates quantity quotas of files and directories.
- The **Space Quota** column indicates storage space sizes of tenant directories.

----End

Add a tenant directory.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click the tenant whose HDFS storage directory needs to be changed.

Step 3 Click **Resource**.

Step 4 In the **HDFS Storage** table, click **Create Directory**.

- The **Parent Directory** indicates the storage directory of the parent tenant corresponding to the current tenant.

  NOTE

  This parameter is not displayed if the current tenant is not a sub-tenant.

- Set **Path** to a tenant directory path.

  NOTE

  If the current tenant is not a sub-tenant, the new path is created in the HDFS root directory.

- Set **Quota** to the quotas of file and directory quantity.
- **File Number Threshold (%)** takes effect only when **Quota** is specified. If the ratio of the number of used files to the value of **Quota** exceeds the value of this threshold, an alarm is generated. If this parameter is not specified, no alarm is reported in this scenario.

  NOTE

  The number of used files is collected every hour. Therefore, the alarm indicating that the file number exceeds the threshold is delayed.

- Set **Space Quota** to storage space sizes of tenant directories.
- **Storage Space Threshold (%)**: If the ratio of used storage space to the value of **Space Quota** exceeds the value of this parameter, an alarm is generated. If this parameter is not specified, no alarm is generated in this scenario.

  NOTE

  The used storage space is collected every hour. Therefore, the alarm indicating that the storage space exceeds the threshold is delayed.

Step 5 Click **OK**.

----End

Modify a tenant directory properties.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click the tenant whose HDFS storage directory needs to be changed.

Step 3 Click **Resource**.

Step 4 In the **HDFS Storage** table, click **Modify** in the **Operation** column of the specified tenant directory.

- Set **Quota** to the quotas of file and directory quantity.
- **File Number Threshold (%)** takes effect only when **Quota** is specified. If the ratio of the number of used files to the value of **Quota** exceeds the value of this threshold, an alarm is generated. If this parameter is not specified, no alarm is reported in this scenario.
- Set **Space Quota** to storage space sizes of tenant directories.
- **Storage Space Threshold (%)**: If the ratio of used storage space to the value of **Space Quota** exceeds the value of this parameter, an alarm is generated. If this parameter is not specified, no alarm is generated in this scenario.

Step 5 Click **OK**.

----End

Delete tenant directory.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click the tenant whose HDFS storage directory needs to be changed.

Step 3 Click **Resource**.

Step 4 In the **HDFS Storage** table, click **Delete** in the **Operation** column of the specified tenant directory.

NOTE

The tenant directory that is created by the system during tenant creation cannot be deleted.

Step 5 Click **OK**.

----End

10.7.2.2.2 Restoring Tenant Data

Scenario

Tenant data is stored on Manager and in cluster components. After components are recovered from faults or reinstalled, some tenant configuration data may be in the abnormal state. Administrator need to manually restore the configuration data on FusionInsight Manager.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click a tenant node.

Step 3 Check the status of the tenant data.

1. In **Summary**, check the color of the circle on the right of **Tenant Status**. Green indicates that the tenant is available and gray indicates that the tenant is unavailable.
2. Click **Resource** and check the color of the circle on the left of **Yarn** or **HDFS Storage**. Green indicates that the resource is available and gray indicates that the resource is unavailable.
3. Click **Service Association** and check the **Status** column of the associated service table. **Normal** indicates that the component can provide services for the associated tenant. **Not Available** indicates that the component cannot provide services for the tenant.
4. If any of the preceding check items is abnormal, go to **Step 4** to restore tenant data.

Step 4 Click . In the dialog box that is displayed, enter the password of the administrator who has logged in for authentication, and click **OK**.

Step 5 In the **Restore Tenant Resource Data** window, select one or multiple components whose data needs to be restored and click **OK**. The system automatically restores the tenant data.

----End

10.7.2.2.3 Deleting a Tenant

Scenario

Based on service requirements, the administrator can delete tenants that are no longer used on FusionInsight Manager to release resources occupied by tenants.

Prerequisites

- A tenant has been added.
- The tenant to be deleted has no sub-tenant.
- The role of the tenant to be deleted is not associated with any user or user group.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, select the tenant to be deleted and click .

NOTE

- If you want to save the tenant data, select **Reserve the data of this tenant resource.**. Otherwise, the tenant's storage space will be deleted.

Step 3 Click **OK** to save the settings.

It takes a few minutes to save the configuration. The tenant is deleted successfully. Roles and the storage space of the tenant are also deleted.

NOTE

After the tenant is deleted, the task queue of the tenant still exists in Yarn. The task queue of the tenant is not displayed on the role management page in Yarn.

----End

10.7.2.3 Managing Resources

10.7.2.3.1 Add a Resource Pool

Scenario

This section describes how to logically divide Yarn cluster nodes to combine multiple NodeManagers into a Yarn resource pool. Each NodeManager belongs to one resource pool only. The Administrator can create a customized resource pool on FusionInsight Manager and add hosts that are not added to other customized resource pools to the newly created resource pool.

The system contains a **default** resource pool by default. All NodeManagers that are not added to customized resource pools belong to this resource pool.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Resource Pool**.

Step 3 Click **Add Resource Pool**.

Step 4 In **Add Resource Pool**, set the properties of the resource pool.

- **Cluster**: Select the name of the cluster to which the resource pool is to be added.
- **Name**: Enter the name of the resource pool. The resource pool name consists of 1 to 50 characters, including digits, letters, or underscores (_), but cannot start with an underscore (_).
- **Resource Label**: Resource label of the resource pool, including letters, digists, underscores (_) or hyphens(-). The value contains 1 to 50 characters and must start with a digit or letter.
- **Resource**: In the host list on the left, select the name of a specified host and click >> to add the selected host to the resource pool. Only hosts in the current cluster can be selected. The host list of a resource pool can be left blank.

  NOTE

  You can select the **Resource** based on the host name, CPU, memory, operating system and platform type.

Step 5 Click **OK** to save the settings.

After the resource pool is created, the system administrator can view the name, type, and members of the resource pool in the resource pool list. Hosts that are

added to the customized resource pool are no longer members of the **default** resource pool.

----End

10.7.2.3.2 Modifying a Resource Pool

Scenario

If hosts in the resource pool need to be adjusted based on service requirements, administrators can modify members in the existing resource pool on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Resource Pool**.

Step 3 Locate the row that contains the specified resource pool in the resource pool list, and click **Edit** in the **Operation** column.

Step 4 In **Edit Resource Pool**, modify Hosts.

- Adding a host: In the host list on the left, select the name of a specified host and click >> to add the selected host to the resource pool.
- Deleting a host: In the host list on the right, select the name of a specified host and click << to delete the selected host from the resource pool. The host list of a resource pool can be left blank.

Step 5 Click **OK** to save the settings.

----End

10.7.2.3.3 Deleting a Resource Pool

Scenario

This section describes how to delete an existing resource pool on FusionInsight Manager.

Prerequisites

- Any queue in the cluster cannot use the resource pool to be deleted as its default resource pool; therefore, cancel the default resource pool before deleting a resource pool. For details, see **Configuring a Queue**.
- Additionally, any queue in the cluster is not allowed to configure the resource distribution policy in the resource pool to be deleted; therefore, clear the policy before deleting a resource pool. For details, see **Clearing Queue Configurations**.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Resource Pool**.

Step 3 Locate the row that contains the specified resource pool in the resource pool list, and click **Delete** in the **Operation** column.

Step 4 In the window that is displayed, click **OK**.

----End

10.7.2.3.4 Configuring a Queue

Scenario

The administrator can modify queue configuration of a specific tenant on FusionInsight Manager based on service requirements.

Prerequisites

Tenants who use the Superior scheduler have been added.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 Click the **Dynamic Resource Plan** tab.

Step 3 Click the **Queue Configuration** tab.

Step 4 Set **Cluster** to the name of the cluster to be operated. In the tenant queue table, click **Modify** in the **Operation** column of the specific tenant queue.

NOTE

- You can also open the queue modification page as follows: Click the target tenant in the tenant list on the left of the **Tenant Resources Management** tab page. In the window that is displayed, click **Resource**. On the page that is displayed, click behind **Queue Configuration (*queue name*)**.
- One queue can be bound to only one non-default resource pool.
- For parameters such as **Max Allocated vCores**, **Max Allocated Memory(MB)**, **Max Running Apps**, **Max Running Apps per User** and **Max Pending Apps**, if the sub-tenant value is **-1**, the parent tenant value can be set to a specific limit. If the parent tenant value is a specific limit, the sub-tenant value can be set to **-1**.
- The values of **Max Allocated vCores** and **Max Allocated Memory(MB)** must be changed to values other than **-1** for the modification to take effect.

Table 10-38 Queue configuration parameters

Parameter	Description
Max Master Shares(%)	Indicates the maximum percentage of resources occupied by all ApplicationMasters in the current queue.
Max Allocated vCores	Indicates the maximum number of cores allocated to a Yarn container in the current queue. The default value is -1, indicating that the value range is not limited.

Parameter	Description
Max Allocated Memory(MB)	Indicates the maximum memory allocated to a Yarn container in the current queue. The default value is -1, indicating that the value range is not limited.
Max Running Apps	Indicates the maximum number of tasks supported by the current queue at one time. The default value is **-1**, which indicates that the number of tasks that can be executed concurrently in the queue is not restricted (same meaning as the parameter value left blank). Value **0** indicates that no task can be executed. The value ranges from -1 to 2147483647.
Max Running Apps per User	Indicates the maximum number of tasks allowed for a user in the current queue at one time. The default value is **-1**, which indicates that the number of tasks that can be executed concurrently in the queue is not restricted (same meaning as the parameter value left blank). Value **0** indicates that no task can be executed. The value ranges from -1 to 2147483647.
Max Pending Apps	Indicates the maximum number of tasks that can be suspended in the current queue at one time. The default value is **-1**, which indicates that the number of tasks that can be suspended concurrently in the queue is not restricted (same meaning as the parameter value left blank). Value **0** indicates that no task can be suspended. The value ranges from -1 to 2147483647.
Resource Allocation Rule	Indicates the rule for allocating resources to different tasks of a user. The rule can be FIFO or FAIR. If a user submits multiple tasks in the current queue and the rule is FIFO, the tasks are executed one by one in sequential order; if the rule is FAIR, resources are evenly allocated to all tasks.
Default Resource Label	Indicates that tasks are executed on a node with a specified resource label.
Active	• ACTIVE: indicates that the current queue can receive and execute tasks. • INACTIVE: indicates that the current queue can receive but cannot execute tasks. Tasks submitted to the queue are suspended.
Open	• OPEN: indicates that the current queue is opened. • CLOSED: indicates that the current queue is closed. Tasks submitted to the queue are rejected.
Migrate Queue Upon Fault	If multi-az availability is enabled for the cluster and an AZ is faulty, set **Migrate Queue Upon Fault** to **TRUE** to submit running queues of the tenant to other AZs.

Step 5 Click **OK**. The queue configuration is complete.

----End

10.7.2.3.5 Configuring the Queue Capacity Policy of a Resource Pool

Scenario

After a resource pool is added, capacity policies of available resources need to be configured for Yarn task queues to ensure the proper running of tasks in the resource pool.

This section describes how to configure the queue policy on FusionInsight Manager. Tenant queues with the Superior scheduler can use resources in different resource pools.

Prerequisites

- You have logged in to FusionInsight Manager.
- You have added a resource pool.
- The task queue is not associated with resource pools of other queues except the default resource pool.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 Click the **Dynamic Resource Plan** tab.

Step 3 Click the **Resource Distribution Policy** tab.

Step 4 Set **Cluster** to the name of the cluster to be operated. In **Resource Pool**, select the specified resource pool.

Step 5 Locate the specified queue in **Resource Allocation**, and click **Modify** in the **Operation** column.

Step 6 On the **Resource Configuration Policy** tab page in **Modify Resource Allocation**, configure the resource allocation policy for the task queue in the resource pool.

- **Weight**: indicates the resources that a tenant can obtain. Its initial value is the same as the minimum resource percentage.
- **Minimum Resource**: indicates the minimum resources that a tenant can obtain.
- **Maximum Resource**: indicates the maximum resources that a tenant can obtain.
- **Reserved Resource**: indicates the resources that are reserved for a tenant and cannot be shared by other tenants.

Step 7 On the **User Policy** tab page in **Modify Resource Allocation**, configure the user policy.

NOTE

defaultUser(built-in) indicates that the policy specified by **defaultUser** is used if a user does not specify a policy. The default policy cannot be deleted.

- Click **Add User Policy** to add a user policy.
  - **Username**: indicates the user name.
  - **Weight**: indicates the resources that a user can obtain.
  - **Max vCores**: indicates the maximum number of virtual cores that a user can obtain.
  - **Max Memory(MB)**: indicates the maximum memory that a user can obtain.
- Click **Modify** in the **Operation** column to modify a user policy.
- Click **Clear** in the **Operation** column to delete a user policy.

Step 8 Click **OK** to save the configuration.

----End

10.7.2.3.6 Clearing Queue Configurations

Scenario

The system administrator can clear queue configurations on FusionInsight Manager if a queue does not need resources from a resource pool or a resource pool needs to be disassociated from a queue. Clearing queue configurations means that the resource capacity policy of a queue in the resource pool is canceled.

Prerequisites

If a queue is to be unbound from a resource pool, you have ensured that the resource pool is not the default resource pool of the queue. For details, see **Configuring a Queue**.

Procedure

Step 1 Log in to FusionInsight Manager portal.

Step 2 Choose **Tenant Resources** > **Dynamic Resource Plan**

Step 3 Set **Cluster** to the name of the cluster to be operated. In **Resource Pool**, select the specified resource pool.

Step 4 Locate the row that contains the specified queue in **Resource Allocation** and click **Clear** in the **Operation** column.

Step 5 In **Clear Queue Configuration**, click **OK** to clear the queue configurations in the current resource pool.

----End

10.7.2.4 Managing Global User Policies

Scenario

If a tenant uses a Superior scheduler, the system can control the policy for a specific user in using the resource scheduler, including:

- Maximum number of running tasks
- Maximum number of suspended tasks
- Default queue

Procedure

- Add a scheduling policy.
  a. On FusionInsight Manager, click **Tenant Resources**.
  b. Click the **Dynamic Resource Plan** tab.
  c. Click **Global User Policy**.

     NOTE

     defaults(default setting) indicates that the default policy is used if a user does not specify the global user policy. The default policy cannot be deleted.

  d. Click **Create Global User Policy**. In the window that is displayed, configure the following parameters:

     - **Cluster**: Select the cluster to be operated.
     - **Username**: indicates the user for whom resource scheduling is controlled. Enter the name of an existing user in the cluster.
     - **Max Running Apps**: indicates the maximum number of tasks that the user can run in the cluster.
     - **Max Pending Apps**: indicates the maximum number of tasks that the user can suspend in the cluster.
     - **Default Queue**: indicates the user queue. Enter the name of an existing queue in the cluster.
- Modify a policy.
  a. On FusionInsight Manager, click **Tenant Resources**.
  b. Click the **Dynamic Resource Plan** tab.
  c. Click **Global User Policy**.
  d. In the row that contains the desired user policy, click **Modify** in the **Operation** column.
  e. After adjusting the parameters, click **OK**.
- Delete a policy.
  a. On FusionInsight Manager, click **Tenant Resources**.
  b. Click the **Dynamic Resource Plan** tab.
  c. Click **Global User Policy**.
  d. In the row that contains the desired user policy, click **Delete** in the **Operation** column.

     In the window that is displayed, click **OK**.

10.7.3 Using the Capacity Scheduler in Multi-Tenant Scenarios

10.7.3.1 Creating Tenants

10.7.3.1.1 Adding a Tenant

Scenario

Based on the resource consumption and isolation plan and requirements of services, administrators can create tenants on FusionInsight Manager to meet actual application scenarios.

Prerequisites

- A tenant name has been planned based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
- Resources to be allocated to the current tenant have been planned to ensure that the sum of capacities of direct sub-tenants at every level cannot exceed the current tenant.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 Click ⊕. On the displayed page, configure tenant properties based on **Table 10-39**.

Table 10-39 Tenant parameters

Parameter	Description
Cluster	Select the cluster for which you want to create a tenant.
Name	• Specifies the name of the current tenant. The value consists of 3 to 50 characters, which can be letters, digits, or underscores (_). • Plan a tenant name based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
Tenant Type	Specifies whether the specified tenant is a leaf tenant. • When **Leaf Tenant** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. • When **Non-leaf Tenant** is selected, the current tenant is not a leaf tenant and sub-tenants can be added to the current tenant.

Parameter	Description
Computing Resource	Specifies the dynamic computing resources for the current tenant. ● When **Yarn** is selected, the system automatically creates a task queue in Yarn and the queue is named the same as the name of the tenant. – A leaf tenant can directly submit tasks to the task queue. – A non-leaf tenant cannot directly submit tasks to the task queue. However, Yarn adds an extra task queue (hidden) named **Default** for the non-leaf tenant to record the remaining resource capacity of the tenant. Actual tasks do not run in this queue. ● When dynamic resources are not **Yarn** resources, the system does not automatically create a task queue.
Default Resource Pool Capacity (%)	Specifies the computing resource usage in the **Default** resource pool of the current tenant. The value ranges from 0 to 100%.
Default Resource Pool Max Capacity (%)	Specifies the maximum computing resource usage in the **Default** resource pool of the current tenant. The value ranges from 0 to 100%.
Storage Resource	Specifies storage resources of the current tenant. ● When **HDFS** is selected, the system automatically allocates storage resources. ● When **HDFS** is not selected, the system does not automatically allocate storage resources.
Quota	Specifies the file and directory quantity quota.
Space Quota	Specifies the used HDFS storage space quota of the current tenant. ● Value range: When Space Quota unit is set to **MB**, this parameter ranges from **1** to **8796093022208**. When Space Quota unit is set to **GB**, this parameter ranges from **1** to **8589934592**. ● This parameter indicates the maximum HDFS storage space that can be used by the tenant, but does not indicate the actual space used. ● If the value is greater than the size of the HDFS physical disk space, the maximum space that can be used is all the HDFS physical disk space.

Parameter	Description
Storage Path	Specifies the HDFS storage directory for a tenant. • The system creates a file folder named after the tenant name in the **/tenant** directory by default. For example, the default HDFS storage directory for **ta1** is **/tenant/ta1**. • When a tenant is created for the first time, the system creates the **/tenant** directory in the HDFS root directory. The storage path is customizable.
Description	Configure the description of the current tenant.

NOTE

During the creation of a tenant, the system automatically creates a corresponding role, the computing resources, and the storage resources.

- The new role has the rights on the computing resources and storage resources. The role and its rights are controlled by the system automatically and cannot be controlled manually under **System > Permission > Role**. The role name is *tenant name_cluster ID*. By default, the cluster ID of the first cluster is not displayed.
- When using this tenant, create a system user and bind the user to the role of the tenant. For details, see **Adding a User and Binding the User to a Tenant Role**.
- During the creation of a tenant, the system automatically creates a Yarn task queue named after the tenant. If the queue name exists, the new queue is named **Tenant name-N**. **N** indicates a natural number starting from 1. When a same name exists, the value **N** increases automatically to differentiate the queue from others. For example, **saletenant**, **saletenant-1**, and **saletenant-2**.

Step 3 Whether the current tenant need to associate resources of other services.

- If yes, go to **4**.
- If no, go to **Step 5**.

Step 4 Click **Associated Service** to configure other service resources used by the current tenant.

1. Select **HBase** in **Service**.
2. Make a selection in **Association Type**:
   - **Exclusive** indicates service resources used by the tenant exclusively. Other tenants cannot associate with this service.
   - **Share** indicates shared service resources, which can be used by other tenants.

NOTE

- When creating a tenant, you can only associate HBase with the tenant. For existing tenants, you can associate the following services: HDFS, HBase, and Yarn.
- Associating existing tenants with service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page and click **Associated Service** to associate the current tenant with service services.
- Canceling the association between existing tenants and service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page, click **Delete**, select **I have read the information and understand the impact.**, and click **OK** to cancel the association with service.

3. Click **OK**.

Step 5 Click **OK**. When **Tenant created successfully.** is displayed on the page, the tenant is added successfully.

----End

10.7.3.1.2 Adding a Sub-Tenant

Scenario

Based on the resource consumption and isolation plan and requirements of services, administrators can create Sub-Tenants on FusionInsight Manager, and allocate resources of the current tenant to meet the actual application scenario.

Prerequisites

- A parent tenant has been added, and belongs to a non-leaf tenant.
- A tenant name has been planned based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
- Resources to be allocated to the current tenant have been planned to ensure that the sum of capacities of direct sub-tenants at every level cannot exceed the current tenant.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, move the cursor to the tenant node to which the sub-tenant is added. Click . In the displayed window, configure the sub-tenant properties based on **Table 10-40**.

Table 10-40 Sub-tenant parameters

Parameter	Description
Cluster	Specifies the cluster of the parent tenant.
Parent Tenant Resource	Specifies the name of the parent tenant.

Parameter	Description
Name	• Specifies the name of the current tenant. The value consists of 3 to 50 characters, which can be letters, digits, or underscores (_). • Plan a sub-tenant name based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
Tenant Type	Specifies whether the specified tenant is a leaf tenant. • When **Leaf Tenant** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. • When **Non-leaf Tenant** is selected, the current tenant is not a leaf tenant and sub-tenants can be added to the current tenant. However, the tenant depth cannot exceed 5 levels.
Computing Resource	Specifies the dynamic computing resources for the current tenant. • When **Yarn** is selected, the system automatically creates a task queue in Yarn and the queue is named the same as the name of the tenant. – A leaf tenant can directly submit tasks to the task queue. – A non-leaf tenant cannot directly submit tasks to the task queue. However, Yarn adds an extra task queue (hidden) named **Default** for the non-leaf tenant to record the remaining resource capacity of the tenant. Actual tasks do not run in this queue. • When dynamic resources are not **Yarn** resources, the system does not automatically create a task queue.
Default Resource Pool Capacity (%)	Specifies the computing resource usage of the current tenant. The base value is the total resources of the parent tenant.
Default Resource Pool Max Capacity (%)	Specifies the maximum computing resource usage of the current tenant. The base value is the total resources of the parent tenant.
Storage Resource	Specifies storage resources of the current tenant. • When **HDFS** is selected, the system automatically creates a file in the HDFS parent tenant directory. The file is named the same as the name of the sub-tenant. • When **HDFS** is not selected, the system does not automatically allocate storage resources.

Parameter	Description
Quota	Specifies the file and directory quantity quota.
Space Quota	Specifies the used HDFS storage space quota of the current tenant. • When **Space Quota Unit** is set to **MB**, this parameter ranges from **1** to **8796093022208**. When **Space Quota Unit** is set to GB, this parameter ranges from **1** to **8589934592**. The maximum value of this parameter does not exceed the total storage quota of the parent tenant. • This parameter indicates the maximum HDFS storage space that can be used by the tenant, but does not indicate the actual space used. • If the value is greater than the size of the HDFS physical disk space, the maximum space that can be used is all the HDFS physical disk space. • If this quota is greater than the quota of the parent tenant, the actual storage space will be affected by the quota of the parent tenant.
Storage Path	Specifies the HDFS storage directory for a tenant. • The system creates a file folder named after the sub-tenant name in the directory of the parent tenant by default. For example, if the sub-tenant is **ta1s** and the parent directory is **/tenant/ta1**, the system sets the **Storage Path** for the sub-tenant to **/tenant/ta1/ta1s**. • The storage path is customizable in the parent directory.
Description	Configure the description of the current tenant.

NOTE

During the creation of a tenant, the system automatically creates a corresponding role, the computing resources, and the storage resources.

- The new role has the rights on the computing resources and storage resources. The role and its rights are controlled by the system automatically and cannot be controlled manually under **System > Permission > Role**. The role name is *tenant name_cluster ID*. By default, the cluster ID of the first cluster is not displayed.
- When using this tenant, create a system user and bind the user to the role of the tenant. For details, see **Adding a User and Binding the User to a Tenant Role**.
- The sub-tenant can further allocate the resources of the current tenant. The sum of the resource percentage of direct sub-tenants of a parent tenant cannot exceed 100%. The sum of the computing resource percentage of all level-1 tenants cannot exceed 100%.

Step 3 Whether the current tenant need to associate resources of other services.

- If yes, go to **Step 4**.

- If no, go to **Step 5**.

Step 4 Click **Associated Service** to configure other service resources used by the current tenant.

1. Select **HBase** in **Service**.
2. Make a selection in **Association Type**:
   - **Exclusive** indicates service resources used by the tenant exclusively. Other tenants cannot associate with this service.
   - **Share** indicates shared service resources, which can be used by other tenants.

NOTE

- When creating a tenant, you can only associate HBase with the tenant. For existing tenants, you can associate the following services: HDFS, HBase, and Yarn.
- Associating existing tenants with service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page and click **Associated Service** to associate the current tenant with service services.
- Canceling the association between existing tenants and service resources: In the tenant list on the left of the **Tenant Management** page, click the target tenant. Then, switch to the **Service Association** tab page, click **Delete**, select **I have read the information and understand the impact.**, and click **OK** to cancel the association with service.

3. Click **OK**.

Step 5 Click **OK**. When **Tenant created successfully.** is displayed on the page, the tenant is added successfully.

----End

10.7.3.1.3 Adding a User and Binding the User to a Tenant Role

Scenario

The created tenant cannot directly log in to the cluster to access resources. Administrators need to create a user for a tenant on FusionInsight Manager and bind the user to a tenant role to assign operation rights to the user.

Prerequisites

The system administrator has understood service requirements and created a tenant.

Procedure

Step 1 On FusionInsight Manager, click **System > Permission > User**.

Step 2 To add a user to the system, click **Create**.

To bind tenant rights to an existing user in the system, click **Modify** in the column where the user locates. The configuration page is displayed.

For details about configuring parameters of a user, see **Table 10-41**.

Table 10-41 User parameters

Parameter	Description
Username	Specifies the name of the current tenant. The value consists of 3 to 32 characters, which can be letters, digits, underlines (_), hyphens(-), or spaces. ● **Username** cannot be the same as any username of the OS on each node in the cluster. Otherwise, the user account cannot be used properly. ● Usernames of the same letters but different cases are not supported. For example, if **User1** already exists, user **user1** cannot be created. When using user **User1**, enter the correct username.
User Type	Options include **Human-Machine** and **Machine-Machine**. ● **Human-Machine** user: Used in FusionInsight Manager O&M scenarios and component client operation scenarios. If you select **Human-Machine**, you need to set **Password** and **Confirm password**. ● **Machine-Machine** user: Used in application development scenarios. If you select **Machine-Machine**, the user password is generated randomly.
Password	If you select **Human-Machine**, set **Password**. The password must contain 8 to 64 characters, consisting at least 4 of uppercase letters, lowercase letters, digits, and special characters and spaces. Cannot be the username or username spelled backwards.
Confirm Password	Enter the password again.
User Group	In **User Group**, click **Add** to add the user to a user group. ● If a role is added to a user group, users in the user group can obtain the rights of the role. ● For example, assign Hive rights to the new user and add the user to the **Hive** group.
Primary Group	Select a group as the primary group of directories and files of the user. The drop-down list contains groups that are selected in **User Group**.
Role	Click **Add** to add a role to the user as required. **NOTE** ● If a user wants to use resources allocated to **tenant1** add sub-tenants to or delete sub-tenants from **tenant1**, bind the **Manager_tenant** and **tenant1**_*cluster ID* roles to the user.

Parameter	Description
Description	Configure the description of the current user.

Step 3 Click **OK**.

----End

10.7.3.2 Managing Tenants

10.7.3.2.1 Managing a Tenant Directory

Scenario

The administrator manages the HDFS storage directory used by a specified tenant on FusionInsight Manager based on service requirements. The management operations include adding tenant directories, modifying quantity quotas of files and directories, and storage space quota of the directory, and deleting directories.

Prerequisites

Tenants with HDFS storage resources are added.

Procedure

View a tenant directory.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click a target tenant.

Step 3 Click **Resource**.

Step 4 View the **HDFS Storage** table.

- The **Quota** column indicates quantity quotas of files and directories.
- The **Space Quota** column indicates storage space sizes of tenant directories.

----End

Add a tenant directory.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click the tenant whose HDFS storage directory needs to be changed.

Step 3 Click **Resource**.

Step 4 In the **HDFS Storage** table, click **Create Directory**.

- The **Parent Directory** indicates the storage directory of the parent tenant corresponding to the current tenant.

  NOTE

  This parameter is not displayed if the current tenant is not a sub-tenant.

- Set **Path** to a tenant directory path.

  NOTE

  If the current tenant is not a sub-tenant, the new path is created in the HDFS root directory.

- Set **Quota** to the quotas of file and directory quantity.
- **File Number Threshold (%)** takes effect only when **Quota** is specified. If the ratio of the number of used files to the value of **Quota** exceeds the value of this threshold, an alarm is generated. If this parameter is not specified, no alarm is reported in this scenario.

  NOTE

  The number of used files is collected every hour. Therefore, the alarm indicating that the file number exceeds the threshold is delayed.

- Set **Space Quota** to storage space sizes of tenant directories.
- **Storage Space Threshold (%)**: If the ratio of used storage space to the value of **Space Quota** exceeds the value of this parameter, an alarm is generated. If this parameter is not specified, no alarm is generated in this scenario.

  NOTE

  The used storage space is collected every hour. Therefore, the alarm indicating that the storage space exceeds the threshold is delayed.

Step 5 Click **OK**.

----End

Modify a tenant directory properties.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click the tenant whose HDFS storage directory needs to be changed.

Step 3 Click **Resource**.

Step 4 In the **HDFS Storage** table, click **Modify** in the **Operation** column of the specified tenant directory.

- Set **Quota** to the quotas of file and directory quantity.
- **File Number Threshold (%)** takes effect only when **Quota** is specified. If the ratio of the number of used files to the value of **Quota** exceeds the value of this threshold, an alarm is generated. If this parameter is not specified, no alarm is reported in this scenario.
- Set **Space Quota** to storage space sizes of tenant directories.
- **Storage Space Threshold (%)**: If the ratio of used storage space to the value of **Space Quota** exceeds the value of this parameter, an alarm is generated. If this parameter is not specified, no alarm is generated in this scenario.

Step 5 Click **OK**.

----End

Delete tenant directory.

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click the tenant whose HDFS storage directory needs to be changed.

Step 3 Click **Resource**.

Step 4 In the **HDFS Storage** table, click **Delete** in the **Operation** column of the specified tenant directory.

NOTE

The tenant directory that is created by the system during tenant creation cannot be deleted.

Step 5 Click **OK**.

----End

10.7.3.2.2 Restoring Tenant Data

Scenario

Tenant data is stored on Manager and in cluster components. After components are recovered from faults or reinstalled, some tenant configuration data may be in the abnormal state. Administrator need to manually restore the configuration data on FusionInsight Manager.

Prerequisites

You have logged in to FusionInsight Manager.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, click a tenant node.

Step 3 Check the status of the tenant data.

1. In **Summary**, check the color of the circle on the right of **Tenant Status**. Green indicates that the tenant is available and gray indicates that the tenant is unavailable.
2. Click **Resource** and check the color of the circle on the left of **Yarn** or **HDFS Storage**. Green indicates that the resource is available and gray indicates that the resource is unavailable.
3. Click **Service Association** and check the **Status** column of the associated service table. **Normal** indicates that the component can provide services for the associated tenant. **Not Available** indicates that the component cannot provide services for the tenant.
4. If any of the preceding check items is abnormal, go to **Step 4** to restore tenant data.

Step 4 Click . In the dialog box that is displayed, enter the password of the administrator who has logged in for authentication, and click **OK**.

Step 5 In the **Restore Tenant Resource Data** window, select one or multiple components whose data needs to be restored and click **OK**. The system automatically restores the tenant data.

----End

10.7.3.2.3 Deleting a Tenant

Scenario

Based on service requirements, the administrator can delete tenants that are no longer used on FusionInsight Manager to release resources occupied by tenants.

Prerequisites

- A tenant has been added.
- The tenant to be deleted has no sub-tenant.
- The role of the tenant to be deleted is not associated with any user or user group.

Procedure

Step 1 On FusionInsight Manager, click **Tenant Resources**.

Step 2 In the tenant list on the left, select the tenant to be deleted and click .

NOTE

- If you want to save the tenant data, select **Reserve the data of this tenant resource.**. Otherwise, the tenant's storage space will be deleted.
- If a user not in the supergroup group is used to delete a tenant and the tenant data is not retained, you need to log in to the HDFS client as a user in the supergroup group and manually clear the storage space of the tenant to avoid residual data.

Step 3 Click **OK** to save the settings.

It takes a few minutes to save the configuration. The tenant is deleted successfully. Roles and the storage space of the tenant are also deleted.

NOTE

After the tenant is deleted, the task queue of the tenant still exists in Yarn. The task queue of the tenant is not displayed on the role management page in Yarn.

----End

10.7.3.2.4 Clearing Unassociated Queues of a Tenant in Capacity Scheduler Mode

Scenario

In Yarn Capacity Scheduler mode, to delete a tenant, set the capacity of a tenant's queue to **0** and the state of the tenant to **STOPPED** to delete a tenant. However, queues of the tenant remain in the Yarn service. Queues cannot be automatically deleted due to the Yarn mechanism. The administrator can run the commands to manually delete the queues.

Impact on the System

- Running the script will restart the controller service, synchronize Yarn configuration, and restart instances of the active and standby ResourceManagers.
- You cannot log in to FusionInsight Manager and perform operations on FusionInsight Manager when restarting the controller service.
- After the instances of the active and standby ResourceManagers are restarted, an alarm will be reported indicating that Yarn and components that depend on Yarn will be temporarily unavailable.

Prerequisites

- You have logged in to FusionInsight Manager.
- Queues of a deleted tenant still exist.

Procedure

Step 1 Check whether queues of the deleted tenant still exist.

1. On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn**. Click **ResourceManager(Active)** to go to the ResourceManager WebUI.
2. Click **Scheduler** on the left pane, and you can see that queues of the tenant still exist. The state of the tenant is STOPPED and **Configured Capacity** is set to **0**.

Step 2 Log in to the active OMS node as user **omm**.

Step 3 Go to the **${BIGDATA_HOME}/om-server/om/sbin** directory and run the **cleanQueuesAndRestartRM.sh** script.

cd ${BIGDATA_HOME}/om-server/om/sbin

./cleanQueuesAndRestartRM.sh -c *Cluster ID*

NOTE

Replace *Cluster ID* with the ID of the cluster to be operated, which can be queried by choosing **Cluster** > *Name of the desired cluster* > **Cluster Properties** on FusionInsight Manager.

During the script execution, enter **yes** and the password.

```
Running the script will restart Controller and restart ResourceManager.
Are you sure you want to continue connecting (yes/no)?yes
Please input admin password:
Begin to backup queues ...
...
```

Step 4 After the script is successfully executed, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** on FusionInsight Manager. Click **ResourceManager(Active)** to go to the ResourceManager WebUI.

Step 5 Click **Scheduler** on the left pane and check whether queues of the deleted tenant are cleared.

----End

10.7.3.3 Managing Resources

10.7.3.3.1 Add a Resource Pool

Scenario

This section describes how to logically divide Yarn cluster nodes to combine multiple NodeManagers into a Yarn resource pool. Each NodeManager belongs to one resource pool only. The Administrator can create a customized resource pool on FusionInsight Manager and add hosts that are not added to other customized resource pools to the newly created resource pool.

The system contains a **Default** resource pool by default. All NodeManagers that are not added to customized resource pools belong to this resource pool.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Resource Pool**.

Step 3 Click **Add Resource Pool**.

Step 4 In **Add Resource Pool**, set the properties of the resource pool.

- **Cluster**: Select the name of the cluster to which the resource pool is to be added.
- **Name**: Enter the name of the resource pool. The resource pool name consists of 1 to 50 characters, including digits, letters, or underscores (_), but cannot start with an underscore (_).
- **Resource**: In the host list on the left, select the name of a specified host and click >> to add the selected host to the resource pool. Only hosts in the current cluster can be selected. The host list of a resource pool can be left blank.

  NOTE

  You can select the **Resource** based on the host name, CPU, memory, operating system and platform type.

Step 5 Click **OK** to save the settings.

After the resource pool is created, the system administrator can view the name, type, and members of the resource pool in the resource pool list. Hosts that are added to the customized resource pool are no longer members of the **Default** resource pool.

----End

10.7.3.3.2 Modifying a Resource Pool

Scenario

If hosts in the resource pool need to be adjusted based on service requirements, administrators can modify members in the existing resource pool on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Resource Pool**.

Step 3 Locate the row that contains the specified resource pool in the resource pool list, and click **Edit** in the **Operation** column.

Step 4 In **Edit Resource Pool**, modify Hosts.

- Adding a host: In the host list on the left, select the name of a specified host and click >> to add the selected host to the resource pool.
- Deleting a host: In the host list on the right, select the name of a specified host and click << to delete the selected host from the resource pool. The host list of a resource pool can be left blank.

Step 5 Click **OK** to save the settings.

----End

10.7.3.3.3 Deleting a Resource Pool

Scenario

This section describes how to delete an existing resource pool on FusionInsight Manager.

Prerequisites

- Any queue in the cluster cannot use the resource pool to be deleted as its default resource pool; therefore, cancel the default resource pool before deleting a resource pool. For details, see **Configuring a Queue**.
- Additionally, any queue in the cluster is not allowed to configure the resource distribution policy in the resource pool to be deleted; therefore, clear the policy before deleting a resource pool. For details, see **Clearing Queue Configurations**.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Resource Pool**.

Step 3 Locate the row that contains the specified resource pool in the resource pool list, and click **Delete** in the **Operation** column.

Step 4 In the window that is displayed, click **OK**.

----End

10.7.3.3.4 Configuring a Queue

Scenario

The administrator can modify queue configuration of a specific tenant on FusionInsight Manager based on service requirements.

Prerequisites

Tenants who use the Capacity scheduler have been added.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Dynamic Resource Plan**.

By default, the **Resource Distribution Policy** tab page is displayed.

Step 3 Click the **Queue Configuration** tab.

Step 4 Set **Cluster** to the name of the cluster to be operated. In the tenant queue table, click **Modify** in the **Operation** column of the specific tenant queue.

NOTE

- You can also open the queue modification page as follows: Click the target tenant in the tenant list on the left of the **Tenant Resources Management** tab page. In the window that is displayed, click **Resource**. On the page that is displayed, click behind the **Queue Configuration (*queue name*)**.
- One queue can be bound to only one non-Default resource pool. A newly added resource pool can be bound to only one queue, which serves as the default resource pool of the queue.

Table 10-42 Queue configuration parameters

Parameter	Description
Tenant Resources Name (Queue)	Tenant name and queue name.
Maximum Application	Specifies the maximum number of applications.
Maximum AM Resource Percent	Specifies the maximum percentage of resources that can be used to run the application master in a cluster.

Parameter	Description
Minimum User Limit Percent(%)	Specifies the minimum resource assurance (percentage) of a user. The resources for each user in a queue are limited at any time. If application programs of multiple users are running at the same time in a queue, the resource usage of each user fluctuates between the minimum value and the maximum value. The minimum value is determined by the number of running application programs, while the maximum value is determined by this parameter. For example, assume that this parameter is set to 25. If two users submit application programs to the queue, each user can use a maximum of 50% resources; if three users submit application programs to the queue, each user can use a maximum of 33% resources; if four users submit application programs to the queue, each user can use a maximum of 25% resources.
User Limit Factor	Specifies the maximum user resource usage limit factor. The maximum user resource usage percentage can be obtained by multiplying the actual resource usage percent of the current tenant in the cluster with this factor.
Status	Specifies the current status of a resource plan. **Running** indicates that the resource plan is running. **Stopped** indicates that the resource plan is stopped.
Default Resource Pool	Specifies the resource pool used by a queue. The default value is **Default**. If you want to change the resource pool, configure the queue capacity first. For details, see **Configuring the Queue Capacity Policy of a Resource Pool**.

Step 5 Click **OK**. The queue configuration is complete.

----End

10.7.3.3.5 Configuring the Queue Capacity Policy of a Resource Pool

Scenario

After a resource pool is added, the capacity policy of available resources needs to be configured for Yarn task queues to ensure proper task running in the resource pool. Each queue can be configured with the queue capacity policy of one resource pool only.

Administrators can View queues in any resource pool and configure the queue capacity policy. After the queue policies are configured, Yarn task queues are associated with resource pools.

Prerequisites

Queues are added, this is, tenants associated to compute resources are created.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Tenant Resources** > **Dynamic Resource Plan**.

By default, the **Resource Distribution Policy** tab page is displayed.

Step 3 Set **Cluster** to the name of the cluster to be operated. In **Resource Pool**, select the specified resource pool.

Step 4 Locate the specified queue in the **Resource Allocation** table, and click **Modify** in the **Operation** column.

Step 5 In **Modify Resource Allocation**, configure the resource capacity policy of the task queue in the resource pool.

- **Resource Capacity (%)**: specifies the computing resource usage percent of the current tenant.
- **Maximum Resource Capacity (%)**: specifies the maximum computing resource usage percent of the current tenant.

Step 6 Click **OK** to save the settings.

NOTE

To cancel the resource capacity policy of the task queue in the resource pool, delete the resource capacity value and save the setting. Then the queue is disassociated with the resource pool. Change the default resource pool of the queue to another resource pool. For details, see **Configuring a Queue**.

----End

10.7.3.3.6 Clearing Queue Configurations

Scenario

The system administrator can clear queue configurations on FusionInsight Manager if a queue does not need resources from a resource pool or a resource pool needs to be disassociated from a queue. Clearing queue configurations means that the resource capacity policy of a queue in the resource pool is canceled.

Prerequisites

If a queue is to be unbound from a resource pool, you have ensured that the resource pool is not the default resource pool of the queue. For details, see **Configuring a Queue**.

Procedure

Step 1 Log in to FusionInsight Manager portal.

Step 2 Choose **Tenant Resources** > **Dynamic Resource Plan**

Step 3 Set **Cluster** to the name of the cluster to be operated. In **Resource Pool**, select the specified resource pool.

Step 4 Locate the row that contains the specified queue in **Resource Allocation** and click **Clear** in the **Operation** column.

Step 5 In **Clear Queue Configuration**, click **OK** to clear the queue configurations in the current resource pool.

----End

10.7.4 Switching the Scheduler

Scenario

The newly installed MRS cluster uses the Superior scheduler by default. If the cluster is upgraded from an earlier version, the administrator can switch the Yarn scheduler from the Capacity scheduler to the Superior scheduler by one click.

Prerequisites

- The network connection for the cluster is proper and secure, and the Yarn service status is normal.
- During scheduler switching, tenants cannot be added, deleted, or modified. In addition, services cannot be started or stopped.

Impact on the System

- Because the Resource Manager is restarted during scheduler switching, submitting tasks to Yarn fails.
- During scheduler switching, tasks in a job being executed on Yarn will continue, but new tasks cannot be started.
- After scheduler switching is complete, tasks on Yarn may fail, causing service interruption.
- After scheduler switching is complete, parameters of the Superior scheduler are used for tenant management.
- After the scheduler is switched, resources in the Superior scheduler cannot be allocated to the tenant queue whose **capacity** is **0** in the Capacity scheduler. As a result, tasks submitted to the tenant queue fail to be executed. You are advised not to set **capacity** of the tenant queue to **0** in the Capacity scheduler.
- After scheduler switching is complete, you cannot add or delete resource pools, Yarn node labels, or tenants during the trial period. If resource pools, Yarn node labels, or tenants are added or deleted, rollback to the Capacity scheduler is not allowed.

  NOTE

  - The recommended trial period for scheduler switching is one week. If resource pools, Yarn node labels, or tenants are added or deleted during this period, the trial period ends immediately.

- Rollback of scheduler switching may cause the loss of partial or all Yarn task information.

Switching from the Capacity scheduler to the Superior scheduler

Step 1 Ensure that the Yarn service status is normal.

1. Log in to FusionInsight Manager as user **admin**.
2. Choose **Cluster** > *Name of the desired cluster* > **Services** and check whether the Yarn service status is normal.

Step 2 Log in to the active OMS node as **user omm**.

Step 3 Switch the scheduler.

The following switching modes are available:

0: The Capacity scheduler is switched to the Superior scheduler, and the Capacity scheduler configurations are converted into the Superior scheduler configurations.

1: Only the Capacity scheduler configurations are converted into the Superior scheduler configurations.

2: Only the Capacity scheduler is switched to the Superior scheduler.

- Mode 0 is recommended if the cluster environment is simple and the number of tenants is less than 20.

  Run the following command:

  sh ${BIGDATA_HOME}/om-server/om/sbin/switchScheduler.sh -c *Cluster ID* **-m 0**

  NOTE

  Replace *Cluster ID* with the ID of the cluster to be operated, which can be queried by choosing **Cluster** > *Name of the desired cluster* > **Cluster Properties** on FusionInsight Manager.

  ```
  Start to convert Capacity scheduler to Superior Scheduler, clusterId=1
  Start to convert Capacity scheduler configurations to Superior. Please wait...
  Convert configurations successfully. Start to switch the Yarn scheduler to Superior. Please wait...
  Switch the Yarn scheduler to Superior successfully.
  ```

- If the cluster environment or tenant information is complex and you need to retain the queue information of the Capacity scheduler on the Superior scheduler, it is recommended that you use mode 1 first to convert the Capacity scheduler configurations. After checking the converted configuration information, use mode 2 to switch the Capacity scheduler to the Superior scheduler.

  a. Run the following command to convert the Capacity scheduler configurations into the Superior scheduler configurations:

     sh ${BIGDATA_HOME}/om-server/om/sbin/switchScheduler.sh -c *Cluster ID* **-m 1**

     ```
     Start to convert Capacity scheduler to Superior Scheduler, clusterId=1
     Start to convert Capacity scheduler configurations to Superior. Please wait...
     Convert configurations successfully.
     ```

  b. Run the following command to switch the Capacity scheduler to the Superior scheduler:

sh ${BIGDATA_HOME}/om-server/om/sbin/switchScheduler.sh -c *Cluster ID* **-m 2**

```
Start to convert Capacity scheduler to Superior Scheduler, clusterId=1
Start to switch the Yarn scheduler to Superior. Please wait...
Switch the Yarn scheduler to Superior successfully.
```

- If you do not need the queue information of the Capacity scheduler, use mode 2.

  a. Log in to FusionInsight Manager and delete all tenants except the default tenant.

  b. Log in to FusionInsight Manager and delete all resource pools except the default resource pool.

  Run the following command to switch the Capacity scheduler to the Superior scheduler:

  sh ${BIGDATA_HOME}/om-server/om/sbin/switchScheduler.sh -c *Cluster ID* **-m 2**

```
Start to convert Capacity scheduler to Superior Scheduler, clusterId=1
Start to switch the Yarn scheduler to Superior. Please wait...
Switch the Yarn scheduler to Superior successfully.
```

NOTE

You can query the scheduler switching logs on the active OMS node.

- **${BIGDATA_LOG_HOME}/controller/aos/switch_scheduler.log**
- **${BIGDATA_LOG_HOME}/controller/aos/aos.log**

----End

10.8 System Configuration

10.8.1 Configuring Permissions

10.8.1.1 Managing Users

10.8.1.1.1 Creating a User

Scenarios

FusionInsight Manager supports 50000 users (including built-in users) at the maximum. By default, only user **admin** has the highest operation rights of FusionInsight Manager. You need to create users on FusionInsight Manager and assign operation rights to the user based on site requirements.

Prerequisites

You have learned service requirements and created roles required by service scenarios.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 On the user list page, click **Create**.

Step 4 Set **Username**. Including digits, letters, underlines (_), hyphens(-), or spaces. It is case insensitive. It cannot be the same as the username in the system or OS.

Step 5 Set **User Type** to either **Human-Machine** or **Machine-Machine**.

- **Human-Machine** user: used in scenarios such as FusionInsight Manager O&M and component operations on a client. If you select this user type, you need to enter a password and confirm the password in **Password** and **Confirm Password** accordingly.
- **Machine-Machine** user: used for component development. If you select this user type, you do not need to enter a password, because the password is randomly generated.

Step 6 In the **User Group** area, click **Add** to add one or more user groups to the list as required.

NOTE

- If the selected user group is bound to a role or a permission policy is configured in Ranger, the user obtains the permission of the corresponding role.
- After FusionInsight Manager is installed, some user groups generated by default have special permissions. Select a correct user group based on the user group description on the GUI.
- If existing user groups cannot be used, click **Create User Group** to create a user group. For details, see **Adding a User Group**.

Step 7 Select a group from all groups added in **User Group** as the primary group for creating directories and files.

The drop-down list contains all the groups added to the **User Group** area.

NOTE

A user can belong to multiple groups (including the primary and secondary groups, only one primary group, and multiple secondary groups). The primary group of a user is set to facilitate maintenance and comply with the permission mechanism of the hadoop community. In addition, the user's primary group and other groups have the same functions in terms of rights control.

Step 8 In the **Role** area, click **Add** to bind a role for each user.

NOTE

- Adding a role when you create a user can specify the user rights.
- When you create a user, if permissions of a user group that is granted to the user cannot meet service requirements, you can assign other created roles to the user. If an existing role cannot be used, click **Create Role** to create a role. For details, see **Adding a Role**.

  The role and rights assignment takes effect about 3 minutes later. If the rights obtained from the user group meet the requirements, you do not need to add a role.
- After the Ranger authentication is enabled for a component, users are granted with all permissions except the permissions of the default user group or role.
- If the user is not added to a user group, or no role is configured for the user, no information is displayed after the user logs in to FusionInsight Manager.

Step 9 Enter information in the **Description** text box as required.

Step 10 Click **OK**. The user is created.

After a **Human-Machine** user is created, it can be used normally only after the initial password is changed. You can log in to FusionInsight Manager as the user and reset the password as prompted.

----End

10.8.1.1.2 Modifying User Information

Scenarios

You can modify user information on FusionInsight Manager, including the user group, primary group, role permission assignment, and user description.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row where the user whose information needs to be modified, click **Modify**.

Modify the parameters based on site-requirements.

NOTE

Changing the user group of a user or modifying the role rights of a user takes effect 3 minutes at most after the operation is performed.

Step 4 Click **OK**.

----End

10.8.1.1.3 Exporting User Information

Scenarios

You can export information about created users on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Click **Export All** to export all user information at a time.

User information contains the following fields: Username, creation time, description, user type (**0** indicates a **Human-Machine** account, **1** indicates a **Machine-Machine** account), primary group, user group list, and roles the user bound to.

Step 4 In the **Save Type** drop-down list, select **TXT** or **CSV**. Click **Export**.

----End

10.8.1.1.4 Locking a User

Scenarios

Users may be suspended for a long time due to service changes. For security purposes, you can lock such users.

You can lock a user by using either of the following methods:

- Automatic lock: You can set the number of consecutive incorrect password attempts in the password policy to lock the users who fail to log in to the system for a specified number of times. For details, see **Configuring Password Policies**.
- Manual lock: You manually lock a user.

This section describes how to lock the account manually. **Machine-Machine** users cannot be locked.

Impact on the System

After a user is locked, you cannot log in to FusionInsight Manager again or perform identity authentication again in the cluster. The locked user can be used only after you manually unlock the user or wait for the lock time to expire.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the user to be locked, and click **Lock**.

Step 4 In the displayed dialog box, select **I have read the information and understand the impact.**, and click **OK**.

----End

10.8.1.1.5 Unlocking a User

Scenarios

You can unlock a user on FusionInsight Manager if the user is locked after the number of login attempts using incorrect passwords exceeds the threshold. Only users created on FusionInsight Manager can be unlocked.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the user to be unlocked, and click **Unlock**.

Step 4 In the displayed dialog box, select **I have read the information and understand the impact.**, and click **OK**.

----End

10.8.1.1.6 Deleting a User

Scenarios

Based on service requirements, you need to delete system users who are no longer used on FusionInsight Manager.

NOTE

- After a user is deleted, the provisioned ticket granting ticket (TGT) is still valid within 24 hours. The user can use the TGT for security authentication and access the system.
- If the name of a new user is the same as that of a deleted user, all owner rights of the deleted user are inherited. You are advised to determine whether to delete the resources owned by the user based on site requirements, for example, files in the HDFS.
- The default user **admin** cannot be deleted.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the user to be deleted, choose **More** > **Delete**.

NOTE

To delete multiple users in batches, select the users to be deleted and click **Delete**.

Step 4 In the displayed dialog box, click **OK**.

----End

10.8.1.1.7 Changing a User Password

Scenarios

For security purposes, the password of a **Human-Machine** user must be changed periodically.

If users have the permission to use FusionInsight Manager, they can change their password on FusionInsight Manager.

If users do not have the permission to use FusionInsight Manager, they can change their passwords on the client.

Prerequisites

- Users have obtained the current password policies from the administrator.
- Users have installed the client on any node in the cluster and obtain the IP address of the node. Contact the administrator to obtain the password of the client installation user.

Changing Passwords Using FusionInsight Manager

Step 1 Log in to FusionInsight Manager.

Step 2 Move the cursor to the username in the upper right corner of the page.

In the displayed dialog box, click **Change Password**.

Step 3 On the displayed page, set **Old Password**, **New Password**, and **Confirm Password**, and click **OK**.

By default, the password must meet the following complexity requirements:

- It must contain at least eight characters.
- The password must contain at least four types of the following characters: Uppercase letters, lowercase letters, digits, spaces, and special characters. The following special characters are supported: `~!@#$%^&*()-_=+|[{}];',<.>/\?
- It must be different from the username or its reverse.
- The password cannot be a common password that is easy to crack.
- It cannot be the same as the password used in the latest *N* times. *N* is the value of **Repetition Rule** in **Configuring Password Policies**.

----End

Changing a Password on the Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to change the password of a system user. This operation takes effect for all servers.

kpasswd *System user name*

For example, if you want to change the password of system user **test1**, run the **kpasswd test1** command.

By default, the password must meet the following complexity requirements:

- It must contain at least eight characters.
- The password must contain at least four types of the following characters: Uppercase letters, lowercase letters, digits, spaces, and special characters. The following special characters are supported: `~!@#$%^&*()-_=+|[{}];',<.>/\?
- It must be different from the username or its reverse.
- The password cannot be a common password that is easy to crack.
- It cannot be the same as the password used in the latest *N* times. *N* is the value of **Repetition Rule** in **Configuring Password Policies**.

NOTE

If an error occurs during the running of the **kpasswd** command, try the following operations:

- Stop the SSH session and start it again.
- Run the **kdestroy** command and then run the **kpasswd** command again.

----End

10.8.1.1.8 Initializing a Password

Scenarios

If a user forgets the password or the public account password needs to be changed periodically, you can initialize the password on FusionInsight Manager. After the password is initialized, the system user needs to change the password upon first login.

NOTE

This operation applies only to **Human-Machine** users.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the user to be initialized, choose **More** > **Initialize Password**. In the displayed dialog box, enter the password of the current login administrator user and click **OK**. In the displayed confirmation dialog box, click **OK**.

Step 4 Enter a password and confirm the password in **New Password** and **Confirm Password** accordingly. Click **OK**.

By default, the password must meet the following complexity requirements:

- It must contain at least eight characters.
- The password must contain at least four types of the following characters: Uppercase letters, lowercase letters, digits, spaces, and special characters. The following special characters are supported: `~!@#$%^&*()-_=+|[{}];',<.>/\?
- It must be different from the username or its reverse.
- The password cannot be a common password that is easy to crack.
- It cannot be the same as the password used in the latest *N* times. *N* is the value of **Repetition Rule** in **Configuring Password Policies**.

----End

10.8.1.1.9 Exporting an Authentication Credential File

Scenario

If a user uses a security mode cluster to develop applications, the keytab file of the user needs to be obtained for security authentication. You can export keytab files on FusionInsight Manager.

NOTE

After a user password is changed, the exported keytab file becomes invalid, and you need to export a keytab file again.

Prerequisites

Before downloading the keytab file of a Human-Machine user, the password of the user must be changed at least once on the Manager portal or a client; otherwise, the downloaded keytab file cannot be used For details, see **Changing a User Password**.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the user whose keytab file needs to be exported, choose **More** > **Download Authentication Credential**, specify the save path after the file is automatically generated, and keep the file properly.

The authentication credential includes the **krb5.conf** file of the Kerberos service.

After the authentication credential file is decompressed, you can obtain the following two files:

- The **krb5.conf** file contains the authentication service connection information.
- The **user.keytab** file contains user authentication information.

----End

10.8.1.2 Managing User Groups

Scenarios

FusionInsight Manager supports 5000 user groups (including built-in user groups) at the maximum. You can create and manage different user groups based on service scenarios on FusionInsight Manager. A user group is bound to a role to obtain operation rights. After a user is added to a user group, the user group can obtain the operation rights of the user group. A user group can be used to classify users and manage multiple users.

Prerequisites

- You have learned service requirements and created roles required by service scenarios.
- Log in to FusionInsight Manager.

Adding a User Group

Step 1 Choose **System** > **Permission** > **User Group**.

Step 2 Above the user group list, click **Create User Group**.

Step 3 Set **Group Name** and **Description**.

A group name consists of letters, digits, underlines (_), hyphens(-), or spaces. A group name can contain 1 to 64 characters. It is case insensitive. It cannot be the same as the group name in the system.

Step 4 In the **Role** area, click **Add** to select a role and add it.

NOTE

- For components (except HDFS and Yarn) for which Ranger authorization has been enabled, the rights of non-default roles on Manager do not take effect. You need to configure Ranger policies to assign rights to user groups.
- If the policy conditions of HDFS and Yarn resource requests in Ranger are not covered, the component ACL rules still take effect.

Step 5 In the **User** area, click **Add** to select a user and add it.

Step 6 Click **OK**.

The user group is created.

----End

Viewing User Group Information

By default, all users are displayed in the user group list. Click the arrow on the left of a specified user group name to view the details about the user group, such as the number of users, users in the group, and roles bound to the user group.

Modifying Information About a User Group

Locate the row that contains the user group to be modified, click **Modify** to modify the information about the user group.

Exporting Information About a User Group

Click **Export All** to export all user groups information at a time. You can export the user group information in TXT or CSV format.

The user group information contains the following fields: user group name, description, user list, and role list

Deleting a User Group

Locate the row that contains the user group to be deleted, and click **Delete**. To delete multiple user groups in batches, select the user groups to be deleted and click **Delete** above the user group list. The user group contains users and cannot be deleted. To delete a user group, delete all users in the user group by modifying the user group, and then delete the user group.

10.8.1.3 Managing Roles

Scenarios

FusionInsight Manager supports 5000 roles (including built-in roles, excluding roles created by tenants) at the maximum. Based on different service requirements, you need to create and manage different roles on FusionInsight Manager and perform authorization management for FusionInsight Manager and components using roles.

Prerequisites

- You have learned service requirements.
- Log in to FusionInsight Manager.

Adding a Role

Step 1 Choose **System** > **Permission** > **Role**.

Step 2 On the displayed page, click **Create Role** and fill in **Role Name** and **Description**.

A role name consists of letters, digits, and underlines (_). A role name can contain 3 to 50 characters. It cannot be the same as the role name in the system.

Step 3 In the **Configure Resource Permission** list, select the cluster whose rights are to be added and select service rights for the role.

When setting rights for a component, enter a resource name in the **Search** text box in the upper right corner and click the search icon to view the search result.

The search scope covers only directories with current permissions. You cannot search subdirectories. Search by keywords supports fuzzy match and is case-insensitive.

NOTE

- For components (except HDFS and Yarn) for which Ranger authorization has been enabled, the rights of non-default roles on Manager do not take effect. You need to configure Ranger policies to assign rights to user groups.
- If the policy conditions of HDFS and Yarn resource requests in Ranger are not covered, the component ACL rules still take effect.
- A maximum of 1000 permissions can be configured for a component at a time.

Step 4 Click **OK**.

----End

Modifying the Role Information

Locate the row that contains the role to be modified and click **Modify**.

Exporting Role Information

Click **Export All** to export all roles information at a time. You can export the information to a TXT or CSV file.

The role information contains the following fields: Role name, description, and the information about whether the role is the default role.

Deleting a Role

Locate the row that contains the role to be deleted, and click **Delete**. To delete multiple roles in batches, select the roles to be deleted and click **Delete** above the role list. Roles cannot be deleted when bound by users. To delete a user group, delete all users in the user group by modifying the user group, and then delete the user group.

Task Example (Creating a Manager Role)

Step 1 Choose **System** > **Permission** > **Role**.

Step 2 On the displayed page, click **Create Role** and fill in **Role Name** and **Description**.

Step 3 In the **Configure Resource Permission** list, click **Manager**. Set the role permission as follows:

Manager permissions:

- Cluster:
  - **view**: view permission for **Cluster** page, view permission for **Alarm** and **Event** page under **O&M > Alarm**.
  - **management**: management permission for **Cluster** and **O&M** page.
- User:
  - **view**: view permission for **Permission** page under **System**.
  - **management**: management permission for **Permission** page under **System**.
- Audit:

  management: management permission for **Audit** page.

- Tenant:

  management: management permission for **Tenant** page, view permission for **Alarm** and **Event** page under **O&M > Alarm**.

- System:

  management: management permission for System page except the **Permission** page, view permission for **Alarm** and **Event** page under **O&M > Alarm**.

Step 4 Click **OK**.

----End

10.8.1.4 Security Policy

10.8.1.4.1 Configuring Password Policies

Scenarios

Based on service security requirements, you can set password security rules, user login security rules, and user locking rules on FusionInsight Manager.

NOTICE

- Modify password policies based on service security requirements, because they involve user management security. Otherwise, security risks may be caused.
- Change the user password after modifying the password policy, and then the new password policy can take effect.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **Security Policy** > **Password Policy**.

Step 3 For details about the parameters for modifying the password policy, see **Table 10-43**.

Table 10-43 Password policy parameters

Parameter	Description
Minimum Password Length	Indicates the minimum number of characters a password contains. The value ranges from **8** to **64**. The default value is **8**.

Parameter	Description
Character Types	Indicates how many character types in the following 5 types a password can contain: uppercase letters, lowercase letters, digits, and special characters (including ~`!?,.:;-_'() {}[]/<>@#$%^&*+|\= and spaces). The value can be **4** or **5**. The default value is **4**, which means that a password can contain uppercase letters, lowercase letters, digits, and the special characters. If you set the parameter to **5**, a password can contain all the five character types mentioned above.
Password Retries	Indicates the number of consecutive wrong password attempts allowed before the system locks the user. The value ranges from **3** to **30**. Default value is **5**.
User Lockup Time (Min)	Indicates the time period during which a user is locked when the user lockout conditions are met. The value ranges from **5** to **120**. Default value is **5**.
Password Validity Period (Day)	Indicates the validity period of a password. The value ranges from **0** to **90**. **0** indicates that the password is permanently valid. The default value is **90**.
Repetition Rule	When modifying a password, you are not allowed to use the password that has been used in the recent *N* times. *N* ranges from 1 to 5, and the default value is 1. This policy applies to only Human-machine users.
Password Expiration Notification Days	Indicates the number of days in advance users are notified that their passwords are about to expire. It is used to notify password expiration in advance. After the value is set, if the difference between the cluster time and the password expiration time is smaller than this value, the user receives password expiration notifications. When logging in to FusionInsight Manager, the user will be notified that the password is about to expire and a message is displayed asking the user to change the password. The value ranges from **0** to *X* (*X* must be set to the half of the password validity period and rounded down). The value **0** indicates that no notification is sent. The default value is **5**.
Interval for Deleting Authentication Failure Records (Min)	Indicates the interval of retaining incorrect password attempts. The value ranges from **0** to **1440**. **0** indicates that incorrect password attempts are permanently retained, and **1440** indicates that incorrect password attempts are retained for one day. Default value is **5**.

Step 4 Click **OK** to save the configurations. Change the user password after modifying the password policy, and then the new password policy can take effect.

----End

10.8.1.4.2 Configuring the Independent Attribute

Scenarios

User **admin** or administrators who are bound to the Manager_administrator role can configure the Independent attribute function on FusionInsight Manager so that common users (all service users in the cluster) can set or cancel their own Independent attributes.

After the Independent attribute switch is turned on, users need to log in and set the Independent attribute.

Restrictions

- Administrators cannot set or cancel the Independent attribute of a user.
- Administrators cannot obtain the authentication credentials of independent users.

Prerequisites

You have obtained the required administrator username and password.

Procedure

Configuring the Independent Attribute Function Switch

Step 1 Log in to FusionInsight Manager as user **admin** or a user bound to the Manager_administrator role.

Step 2 Choose **System** > **Permission** > **Security Policy** > **Configuration Independent**.

Step 3 Open or Close the **Independent Attribute**, enter the password as prompted and click **OK**.

Step 4 After the authentication succeeds, and the OMS configuration is modified, click **Finish**.

NOTE

After the Independent attribute function switch is closed:

- A user who has the attribute can cancel it by moving the cursor to the username in the upper right corner of the page and choose **Cancel Independent** from the displayed shortcut menu. After the cancellation, the user cannot set the attribute again. After the attribute is cancelled, existing independent tables will retain the attribute. However, the user cannot create independent tables again.
- Users without this attribute cannot set or cancel the attribute.

Configuring the Independent Attribute

Step 5 Log in to FusionInsight Manager as a user.

NOTICE

After the Independent attribute is set by a user, administrators cannot initialize the password of the user. If the user password is forgotten, the password cannot be retrieved.

User **admin** cannot set the Independent attribute.

Step 6 Move the cursor to the username in the upper right corner of the page.

Step 7 Choose **Set Independent** or **Cancel Independent** from the displayed shortcut menu.

NOTE

- If the Independent attribute function switch is turned on, and the attribute of the user is set, **Cancel Independent** is displayed in the shortcut menu.
- If the Independent attribute function switch is turned on, and the attribute of the user is cancelled, **Set Independent** is displayed in the shortcut menu.
- If the Independent attribute function switch is turned off, and the attribute of the user is set, **Cancel Independent** is displayed in the shortcut menu.
- If the Independent attribute function switch is turned off, and the attribute of the user is cancelled, no options are displayed in the shortcut menu.

Step 8 Enter the password as prompted and click **OK**.

Step 9 After the authentication succeeds, click **OK** in the confirmation dialog box.

----End

10.8.2 Configuring Interconnections

10.8.2.1 Configuring SNMP Northbound Parameters

Scenarios

If users need to view alarms and monitoring data of a cluster on the O&M platform, you can use the simple network management protocol (SNMP) on FusionInsight Manager to report related data to the network management system (NMS).

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Interconnection** > **SNMP**.

Step 3 Turn on the switch on the right of **SNMP Service**.

The SNMP service is disabled by default. indicates that the service is enabled.

Step 4 Set interconnection parameters based on the information provided in **Table 10-44**.

Table 10-44 Interconnection parameters

Parameter	Description
Version	Specifies the version of the SNMP, which can be: ● **V2C**: This is an earlier version with low security. ● **V3**: This is a higher version with higher security **V3** is recommended.
Local Port	Specifies the local port. The default value is **20000**. The value ranges from **1025** to **65535**.
Read Community Name	This parameter is available only when **Version** is set to **V2C**. It is used to set the read-only community name.
Write Community Name	This parameter is available only when **Version** is set to **V2C**. It is used to set the write-only community name.
Security Username	This parameter is available only when **Version** is set to **V3**. It is used to set the protocol security username.
Authentication Protocol	This parameter is available only when **Version** is set to **V3**. It is used to set the authentication protocol. SHA is recommended.
Authentication Password	This parameter is available only when **Version** is set to **V3**. It is used to set the authentication key.
Confirm Password	This parameter is available only when **Version** is set to **V3**. It is used to confirm the authentication key.
Encryption Protocol	This parameter is available only when **Version** is set to **V3**. It is used to set the encryption protocol. AES256 is recommended.
Encryption Password	This parameter is available only when **Version** is set to **V3**. It is used to set the encryption key.
Confirm Password	This parameter is available only when **Version** is set to **V3**. It is used to confirm the encryption key.

NOTE

- The value of **Security Username** cannot contain repeated character strings with the unit length is a common factor of 64 (such as 1, 2, 4, and 8), for example, **abab** and **abcdabcd**.
- The authentication password and encryption password must contain 8 to 16 characters, including at least three types of the following characters: uppercase letters, lowercase letters, digits, and special characters. The two passwords must be different. The two passwords cannot be the same as the security username or the reverse of the security username.
- For security purposes, you need to periodically change the authentication and encryption passwords when using SNMP.
- If SNMPv3 is used, a security user will be locked after five consecutive authentication failures within 5 minutes. The user will be automatically unlocked 5 minutes later.

Step 5 Click **Create Trap Target**. On the displayed dialog box, set the following parameters:

- **Target Symbol**: specifies the trap target ID, which is the ID of the NMS or host that receives traps. The value consists of 1 to 255 characters, including letters or digits.
- **Target IP Address Mode**: specifies the IP address mode of the destination IP address. This parameter can be set to **IPV4** or **IPV6**.
- **Target IP Address**: specifies the IP address of the target trap. Destination IP address, which must be able to communicate with the management plane IP address of the management node.
- **Target Port**: specifies the port receiving traps. The port number must be consistent with the peer end and ranges from 0 to 65535.
- **Trap Community Name**: This parameter is available only when **Version** is set to **V2C** and is used to report the community name.

Click **OK**.

The **Create Trap Target** dialog box is closed.

Step 6 Click **OK** to complete the settings.

----End

10.8.2.2 Configuring Syslog Northbound Parameters

Scenarios

If users need to view alarms and events of a cluster on the unified alarm reporting platform, you can use the Syslog protocol on FusionInsight Manager to report related data to the alarm platform.

NOTICE

If the Syslog protocol is not encrypted, data may be stolen.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Interconnection** > **Syslog**.

Step 3 Turn on the switch on the right of **Syslog Service**.

The Syslog service is disabled by default. indicates that the service is enabled.

Step 4 Set northbound parameters based on information provided in **Table 10-45**.

Table 10-45 Syslog interconnection parameters

Parameter Area	Parameter	Description
Syslog Protocol	Server IP Address Mode	Set the IP address mode of the interconnected server.The value can be IPv4 or IPv6.
	Server IP Address	Sets the IP address of the interconnection server.
	Server Port	Specifies the port number for interconnection.
	Protocol	Sets the protocol type. The available values are as follows: • **TCP** • **UDP**
	Severity Level	Specifies the severity of the reported message. The options are as follows: • **Emergency** • **Alert** • **Critical** • **Error** • **Warning** • **Notice** • **Informational** (default value) • **Debug** **NOTE** **Severity** and **Facility** determine the priority of the sent message. **Priority** = **Facility** × 8 + **Severity** For details about the values of **Severity** and **Facility**, see **Table 10-46**.
	Facility	Sets the module where the log is generated. For details about the available value of this parameter, see **Table 10-46**. The default value **local use 0 (local0)** is recommended.

Parameter Area	Parameter	Description
	Identifier	Sets the product ID. The default value is **FusionInsight Manager**. The identifier can contain a maximum of 256 characters, including letters, digits, underscores (_), spaces, |, $, {, }, periods (.), and hyphens (-).
Report Message	Report Format	Sets the message format of the alarm report. For details, see help information on the page. The packet can contain letters, digits, underscores (_), spaces, |, $, {, }, periods (.), and hyphens (-), and cannot exceed 1024 characters. **NOTE** For details about the information field in the packet format, see **Table 10-47**.
	Alarm Type	Sets the type of the alarm to be reported.
	Alarm Severities	Sets the level of the alarm to be reported.
Uncleared Alarm Reporting	Periodic Uncleared Alarm Reporting	Sets whether to report uncleared alarms in a specified period. Turn on the switch indicates that the function is enabled, and turn off the switch indicates that the function is disabled. The switch is turned off by default.
	Report Interval (min)	Sets the interval at which alarms are reported periodically. This parameter is valid only when the switch is turned on on the right of **Periodic Uncleared Alarm Reporting**. The default value is **15**, in minutes. The value ranges from 5 to 1440 (one day).
Heartbeat Settings	Heartbeat Reporting	Sets whether to periodically report Syslog heartbeat messages. Turn on the switch indicates that the function is enabled, and turn off the switch indicates that the function is disabled. The switch is turned off by default.
	Heartbeat Period (min)	Sets the interval at which heartbeat messages are periodically reported. This parameter is valid only when the switch is turned on on the right of **Heartbeat Reporting**. The default value is 15, in minutes. The value ranges from 1 to 60.
	Heartbeat Packet	Sets the heartbeat message to be reported. This parameter is enabled if the switch is turned on on the right of **Heartbeat Reporting**, and cannot be left blank. The value can contain a maximum of 256 characters, including digits, letters, underscores (_), vertical bars (|), colons (:), spaces, commas (,), and periods (.).

NOTE

After the periodic heartbeat packet function is enabled, packets may be interrupted during automatic recovery of some cluster error tolerance (for example, active/standby OMS switchover). In this case, wait for automatic recovery.

Step 5 Click **OK** to complete the settings.

----End

Related Information

Table 10-46 Numeric codes of **Severity** and **Facility**

Severity	Facility	Numeric Code
Emergency	kernel messages	0
Alert	user-level messages	1
Critical	mail system	2
Error	system daemons	3
Warning	security/authorization messages (note 1)	4
Notice	messages generated internally by syslogd	5
Informational	line printer subsystem	6
Debug	network news subsystem	7
-	UUCP subsystem	8
-	clock daemon (note 2)	9
-	security/authorization messages	10
-	FTP daemon	11
-	NTP subsystem	12
-	log audit (note 1)	13
-	log alert (note 1)	14
-	clock daemon	15
-	local use 0~7 (local0 ~ local7)	16 to 23

Table 10-47 Packet format information field

Information Field	Description
dn	Cluster name
id	Alarm ID
name	Alarm name
serialNo	Alarm serial number **NOTE** The sequence numbers of the fault alarms and the corresponding recovery alarms are the same.
category	Alarm type. The options are as follows: ● **0**: fault alarms ● **1**: cleared alarms ● **2**: event
occurTime	Time when the alarm was generated
clearTime	Time when this alarm is cleared
isAutoClear	Whether an alarm is automatically cleared. The options are as follows: ● **1**: yes ● **0**: no
locationInfo	Location where the alarm is generated
clearType	Alarm clearance type. The options are as follows: ● **-1**: not cleared ● **0**: automatic cleared ● **2**: manually cleared
level	Severity. The options are as follows: ● **1**: critical alarms ● **2**: major alarms ● **3**: minor alarms ● **4**: warning alarms
cause	Alarm cause
additionalInfo	Additional information
object	Alarm object

10.8.2.3 Configuring Monitoring Indicator Data Dump

Scenarios

The monitoring data reporting function writes the monitoring data collected in the system into a text file and uploads the file to a specified server in FTP or SFTP mode.

Before using this function, you need to perform related configurations on FusionInsight Manager.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Interconnection** > **Upload Performance Data**.

Step 3 Click the switch on the right of **Upload Performance Data**.

Upload Performance Data is disabled by default.  indicates the function is enabled.

Step 4 Set the upload parameters based on information provided in **Table 10-48**.

Table 10-48 Uploading configuration parameters

Parameter	Description
FTP IP Address Mode	Specifies the server IP address mode. This parameter is mandatory.The value can be IPv4 or IPv6.
FTP IP Address	Specifies the FTP server for storing monitoring files after the monitoring indicator data is interconnected. This parameter is mandatory.
FTP Port	Specifies the port connected to the FTP server. This parameter is mandatory.
FTP Username	Specifies the username for logging in to the FTP server. This parameter is mandatory.
FTP Password	Specifies the password for logging in to the FTP server. This parameter is mandatory.
Save Path	Specifies the path for storing monitoring files on the FTP server. This parameter is mandatory.
Dump Interval (s)	Specifies the interval at which monitoring files are periodically stored on the FTP server, in seconds. This parameter is mandatory.
Dump Mode	Specifies the protocol used for sending monitoring files. This parameter is mandatory. The available values are **FTP** and **SFTP**. You are advised to use the SFTP mode based on SSH Version 2 (SSHv2). Otherwise, security risks may exist.

Parameter	Description
SFTP Service Public Key	Specifies the public key of the FTP server. This parameter is optional. This parameter is valid only when **Dump Mode** is set to **SFTP**.

Step 5 Click **OK** to complete the settings.

NOTE

In the scenario where the dump mode SFTP is used, if the public key of the SFTP service is empty, the system displays a security risk warning. Determine the security risk, and then save the configuration.

----End

Data Format

After the configuration is complete, the monitoring data reporting function periodically writes monitoring data in the cluster to text files and reports the files to the corresponding FTP/SFTP service based on the configured reporting period.

- Principles for Generating Monitoring Files
  - The monitoring indicators are written to files generated every 30, 60, and 300 seconds based on the indicator collection period.

    30s: indicates real-time indicators whose default collection period is 30s.

    60s: indicates real-time indicators whose default collection period is 60s.

    300s: indicates all other indicators except the ones whose period is 30s or 60s.
  - File name format: **metirc**_{*Period*}_{*File creation time YYYYMMDDHHMMSS* }**.log**

    For example: **metric_60_20160908085915.log**

    metric_300_20160908085613.log
- Monitoring File Contents
  - Format of monitoring files:

    cluster ID |cluster name |indicator ID|collection time|collection host|unit| indicator value. Separate the fields from each other using vertical bars (|), for example:

    ```
    1|xx1|Host|Host|10000413|2019/06/18 10:05:00|189-66-254-146|KB/s|309.910
    1|xx1|Host|Host|10000413|2019/06/18 10:05:00|189-66-254-152|KB/s|72.870
    2|xx2|Host|Host|10000413|2019/06/18 10:05:00|189-66-254-163|KB/s|100.650
    ```

    Note: The actual files are not in the format.
  - Interval for uploading monitoring files:

    The interval for uploading monitoring files can be set using the **Dump Interval (s)** parameter on the page. Currently, the interval can be set to 30s to 300s. After the configuration is complete, the system periodically uploads files to the corresponding FTP/SFTP server at the specified interval.
- Monitoring Indicator Description File

- Indicator set file

  The indicator set file **all-shown-metric-zh_CN** contains detailed information about all indicators. After obtaining the indicator IDs from the files reported by the third-party system, you can query details about the indicators from the indicator set file.

  Location of the indicator set file:

  Active and standby OMS nodes: {*FusionInsight installation path*} **/om-server/om/etc/om/all-shown-metric-zh_CN**

  Contents of the indicator set file:

  ```
  Real-time indicator ID, 5-minute indicator ID, indicator name, indicator collection period (second), whether to collect data by default, service to which the indicator belongs, and role to which the indicator belongs
  00101,10.000101,JobHistoryServer non-heap memory usage,30,false,Mapreduce,JobHistoryServer
  00102,10.000102,JobHistoryServer Non-heap memory allocation volume,30,false,Mapreduce,JobHistoryServer
  00103,10.000103,JobHistoryServer heap memory usage,30,false,Mapreduce,JobHistoryServer
  00104,10.000104,JobHistoryServer heap memory allocation volume,30,false,Mapreduce,JobHistoryServer
  00105, 10.000105,Number of blocked threads,30,false,Mapreduce,JobHistoryServer
  00106,10.000106,Number of running threads,30,false,Mapreduce,JobHistoryServer
  00107,10.000107,GC time,30,false,Mapreduce,JobHistoryServer
  00110,10.00011,JobHistoryServer CPU usage,30,false,Mapreduce,JobHistoryServer
  Real-Time Metric ID,5-Minute Metric ID,Metric Name,Metric Collection Period (s),Collected by Default,Service Belonged To,Role Belonged To 00101,10000101,Used Non Heap Memory of JobHistoryServer,60,false,Mapreduce,JobHistoryServer
  00102,10000102,Allocated Non Heap Memory of JobHistoryServer,60,false,Mapreduce,JobHistoryServer
  00103,10000103,Used Heap Memory of JobHistoryServer,60,false,Mapreduce,JobHistoryServer
  00104,10000104,Allocated Heap Memory of JobHistoryServer,60,false,Mapreduce,JobHistoryServer
  00105,10000105,Blocked Threads,30,false,Mapreduce,JobHistoryServer
  00106,10000106,Running Threads,30,false,Mapreduce,JobHistoryServer
  00107,10000107,GC Time,60,false,Mapreduce,JobHistoryServer
  ```

- Field description of critical indicators

  Real-Time Metric ID: indicates the ID of the indicator whose collection period is 30s or 60s.

  5-Minute Metric: The ID of a 5-minute (300s) indicator.

  Metric Collection Period (s): Real-time collection period of indicators. The value can be **30** or **60**.

  Service Belonged To: Name of the service to which an indicator belongs, indicating the service type, for example, HDFS and HBase.

  Role to which an indicator belongs: indicates the name (type) of the role to which an indicator belongs, for example, JobServer or RegionServer.

- Description

  For metrics whose collection period is 30s/60s, you can find the corresponding metric description by referring to the first column, that is, **Real-Time Metric ID**.

  For metrics whose collection period is 300s, you can find the corresponding metric description by referring to the second column, that is, **5-Minute Metric**.

10.8.3 Importing a Certificate

Scenarios

CA certificates are used to encrypt data during communication for FusionInsight Manager modules, component clients of the cluster, and component servers of the cluster to implement secure communication. CA certificates can be quickly imported to FusionInsight Manager for product security. Import CA certificates in following scenarios:

- When the cluster is installed for the first time, you need to replace the enterprise certificate.
- If the enterprise certificate has expired or security hardening is required, you need to replace it with a new certificate.

Impact on the System

- During certificate replacement, the cluster needs to be restarted. In this case, the system cannot be accessed and cannot provide services.
- After the certificate is replaced, the certificates used by all components and FusionInsight Manager modules are automatically updated.
- After the certificate is replaced, you need to reinstall the certificate in the local environment where the certificate is not trusted.

Prerequisites

- The certificate file and key file can be applied for from the enterprise certificate administrator or generated by the administrator.
- Obtain the files to be imported to the MRS cluster, including the CA certificate file (such as *.crt), key file (*.key), and file (password.property) that saves the key file password. The certificate name and key name can contain uppercase letters, lowercase letters, and digits. After the preceding files are generated, they need to be compressed into a package in TAR format.
- Prepare a password for accessing the key file.

  The password complexity requirements are as follows. If the password complexity does not meet the following requirements, security risks may exist:

  - It must contain at least eight characters.
  - It must contain at least four of the following character types: uppercase letters, lowercase letters, digits, and special characters ~`!?,.;-_'(){}[]/<>@#$%^&*+|\=.
- When applying for a certificate from the certificate administrator, provide the password for accessing the key file and apply for the certificate files in CRT, CER, CERT, and PEM formats and the key files in KEY and PEM formats. The applied certificates must have the issuing function.

Procedure

Step 1 Log in to FusionInsight Manager and choose **System** > **Certificate**.

Step 2 Click ··· on the right of **Upload Certificate**. In the File selection dialog box, view the obtained TAR package of the certificate file and select the file.

Step 3 Click **Upload**.

The system uploads the compressed package and automatically imports the package.

Step 4 After the package is imported, the system prompts you to synchronize the configuration and restart the web service for the new certificate to take effect. Click **OK**.

Step 5 In the displayed dialog box, enter the password of the current login user and click **OK**. The cluster configuration is automatically synchronized and the web service is restarted.

Step 6 After the cluster is restarted, enter the URL for accessing FusionInsight Manager in the address box of the browser and check whether the FusionInsight Manager WebUI can be successfully displayed.

Step 7 Log in to FusionInsight Manager.

Step 8 Choose **Cluster** > *Name of the desired cluster* > **Dashboard** > **More** > **Restart**.

Step 9 In the displayed dialog box, enter the password of the current login user and click **OK**.

----End

10.8.4 OMS Management

10.8.4.1 Overview of the OMS Maintenance Page

Overview

Log in to FusionInsight Manager, choose **System** > **OMS**. On the displayed OMS maintenance page, you can perform maintenance operations on the OMS, including viewing basic information, viewing the service status of OMS service modules, and manually triggering health checks.

Basic Information

OMS-associated information is displayed on FusionInsight Manager, as listed in **Table 10-49**.

Table 10-49 OMS information

Item	Description
Version	Indicates the OMS version, which is consistent with the FusionInsight Manager version.
IP Mode	Indicates the IP address mode of the current cluster network.
HA Mode	Indicates the OMS working mode, which is specified by the configuration file during FusionInsight Manager installation.

Item	Description
Current Active	Indicates the host name of the active OMS node, that is, the host name of the active management node. Click a host name to go to the host details page.
Current Standby	Indicates the host name of the standby OMS node, that is, the host name of the standby management node. Click a host name to go to the host details page.
Duration	Indicates the duration for starting the OMS process.

OMS Service Status

FusionInsight Manager displays the running status of all OMS service modules. If the status of each service module is displayed as ●, the OMS is running properly.

Health check

You can click **Health Check** on the OMS maintenance page to check the OMS status. If some check items are faulty, you can open the check description for troubleshooting.

Entering or Exiting Maintenance Mode

Configure OMS to enter or exit the maintenance mode.

System Parameters

In the large cluster scenario, connect to the DMPS cluster.

10.8.4.2 Modifying OMS Service Configuration Parameters

Scenario

Based on the security requirements of the user environment, you can modify the Kerberos and LDAP configurations in the OMS on FusionInsight Manager.

Impact on the System

After the OMS service configuration parameters are modified, the corresponding OMS module needs to be restarted. In this case, FusionInsight Manager cannot be used.

Procedure

Modifying the okerberos configuration

Step 1 Log in to FusionInsight Manager and choose **System** > **OMS**.

Step 2 Locate the row that contains the okerberos, click **Modify Configuration**.

Step 3 Modify the parameters based on information provided in **Table 10-50**.

Table 10-50 okerberos parameter configuration

Parameter	Description
KDC Timeout (ms)	Timeout duration for an application to connect to the Kerberos, in milliseconds. The value must be an integer.
Max. Retries	Maximum number of attempts for the connection between an application to the Kerberos, in seconds. Set the parameter to an integer.
LDAP Timeout (ms)	Timeout interval for the Kerberos to connect to the LDAP, in milliseconds.
LDAP Search Timeout (ms)	Timeout duration for Kerberos to query user information in the LDAP, in milliseconds.
Kadmin Listening Port	Port number of the kadmin service.
KDC Listening Port	Port number of the kinit service.
Kpasswd Listening Port	Port number of the kpasswd service.

Step 4 Click **OK**.

In the displayed dialog box, enter the password of the current login user and click **OK**. In the displayed confirmation dialog box, click **OK**.

Modifying the oldap configuration

Step 5 Locate the row that contains the on-line data processor (OLDAP), click **Modify Configuration**.

Step 6 Modify the parameters based on information provided in **Table 10-51**.

Table 10-51 OLDAP parameter configuration

Parameter	Description
LDAP Listening Port	Port number of the LDAP service.

Step 7 Click **OK**.

In the displayed dialog box, enter the password of the current login user and click **OK**. In the displayed confirmation dialog box, click **OK**.

NOTE

To reset the password of the LDAP account, you need to restart ACS. The procedure is as follows:

1. Log in to the active management node as user **omm** using PuTTY, and run the following command to update the domain configuration:

   sh ${BIGDATA_HOME}/om-server/om/sbin/restart-RealmConfig.sh

   The command is run successfully if the following information is displayed:

   ```
   Modify realm successfully. Use the new password to log into FusionInsight again.
   ```

2. Run the **sh $CONTROLLER_HOME/sbin/acs_cmd.sh stop** command to stop ACS.
3. Run the **sh $CONTROLLER_HOME/sbin/acs_cmd.sh start** command to start ACS.

Restarting the cluster

Step 8 Log in to FusionInsight Manager and restart the cluster by referring to **Performing a Rolling Restart of a Cluster**.

----End

10.8.5 Component Management

10.8.5.1 Viewing Component Packages

Scenarios

A complete MRS cluster consists of multiple component packages. Before installing some services on FusionInsight Manager, check whether the component packages corresponding to the services have been installed.

Procedure

Step 1 Log in to FusionInsight Manager and choose **System** > **Component**.

Step 2 On the **Installed Component** tab page, view all components.

NOTE

You can view the registered OS and platform type in the **Platform Type** column.

Step 3 Click ⌄ on the left of a component name to view the services and version numbers contained in the component.

----End

10.9 Cluster Management

10.9.1 Configuring Client

10.9.1.1 Installing a Client

Scenario

Install the clients of all services (except Flume). MRS provides shell scripts for different services for maintenance personnel to log in to related maintenance clients and implement maintenance operations.

NOTE

- Reinstall the client after server configuration is modified on the Manager portal or after the system is upgraded. Otherwise, the versions of the client and server will be inconsistent.

Prerequisites

- The client installation directory is automatically created if it does not exist. If it already exists, it must be empty. The directory cannot contain any space.
- If the node where the client is to be installed is a server outside the cluster, it must be able to communicate with the service plane. Otherwise, the client will fail to be installed.
- The client must be enabled with the NTP service and synchronize time with the server. Otherwise, the client will fail to be installed.
- The HDFS and MapReduce components are stored in the same directory (*client directory*/HDFS/) after being downloaded.
- You can install or use the client as any user. Obtain the username and password from the administrator. This section uses user **user_client** as an example. User **user_client** is the owner of the server file directory (such as **/opt/Bigdata/client**) and the client installation directory (such as **/opt/Bigdata/hadoopclient**) with the permissions of **755**.
- You have obtained the component service user (default user or new user) and password.
- If the **/var/tmp/patch** directory already exists when you install the client as non-**root** or non-**omm** user, change the permission on the directory to **777** and the permission on the logs in the directory to **666**.

Procedure

Step 1 Obtain the software package.

Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Click the wanted cluster from the **Cluster** drop-down list.

Choose **More** > **Download Client**. The **Download Cluster Client** window is displayed.

NOTE

In a single-client scenario, choose **Cluster** > *Name of the desired cluster* > **Services** > *Service name* > **More** > **Download Client**. The **Download Client** dialog box is displayed.

Step 2 Set **Select Client type** to **Complete Client**.

Configuration Files Only is to download client configuration files in the following scenario: After all clients are downloaded and installed and administrators modify server configuration on the Manager portal, development personnel need to update the configuration files during application development.

There are two client software packages:

- x86_64: client software package that can be deployed on the x86 platform.
- aarch64: client software package that can be deployed on the TaiShan platform.

NOTE

The cluster supports **x86_64** and **aarch64** clients. However, the client type must match the architecture of the target node. Otherwise, the client installation will fail.

Step 3 Determine whether to generate a client file on the cluster node?

- If yes, select **Save to Path**, and click **OK** to generate the client file. By default, the client file is generated in **/tmp/FusionInsight-Client** on the active management node. The directory can be customized and user **omm** has the read, write, and execute permission on the directory. Click **OK**, copy the software package to the file directory, for example, **/opt/Bigdata/client**, on the server where the client is to be installed as user **omm** or **root**. Then, go to **Step 5**.

  NOTE

  If you cannot obtain permissions of user **root**, use the **omm** user.

- If no, click **OK**, specify a local save path, and download the complete client. Wait until the download is complete, and go to **Step 4**.

Step 4 Upload the software package. Use WinSCP to upload the software package to the server file directory where the client is to be installed (such as **/opt/Bigdata/client**) as the user who is to install the client (any user, such as user **user_client**).

The format of the client software package name is as follows: **FusionInsight_Cluster_***<Cluster ID>*_**Services_Client.tar**. The following steps and sections use **FusionInsight_Cluster_1_Services_Client.tar** as an example.

NOTE

- The host where the client is to be installed can be a node in the cluster or outside the cluster. If the node is a server outside the cluster, it must be able to communicate with the cluster, and the NTP service must be enabled to ensure that the time is the same as that on the server.
- For example, you can configure NTP clock sources for external client servers as well as clusters. Then you can execute the **ntpq -np** command to check whether the time is synchronized.
  - If there is a ***** before the result of the NTP clock source IP address, it means time synchronization is normal, as follows:

    ```
    remote refid st t when poll reach delay offset jitter
    ==============================================================================
    *10.10.10.162 .LOCL. 1 u 1 16 377 0.270 -1.562 0.014
    ```

  - If there is no ***** before the result of the NTP clock source IP address, and the result of **refid** is **.INIT.**, or the results showed abnormal, it means synchronization is exception, please contact technical support.

    ```
    remote refid st t when poll reach delay offset jitter
    ==============================================================================
    10.10.10.162 .INIT. 1 u 1 16 377 0.270 -1.562 0.014
    ```

- You can also configure the same chrony clock source for external servers as that for the cluster. After the configuration, run the **chronyc sources** command to check whether the time is synchronized.
  - In the command output, if an asterisk (*) exists before the IP address of the chrony service on the active OMS node, the synchronization is in normal state. For example:

    ```
    MS Name/IP address         Stratum Poll Reach LastRx Last sample
    ===============================================================================
    ^* 10.10.10.162              10  10   377   626    +16us[  +15us] +/-  308us
    ```

  - If there is no asterisk (*) before the IP address of the NTP service on the active OMS node and **Reach** is "0", the synchronization is abnormal.

    ```
    MS Name/IP address         Stratum Poll Reach LastRx Last sample
    ===============================================================================
    ^? 10.1.1.1                  0  10     0     -     +0ns[   +0ns] +/-    0ns
    ```

Step 5 Log in to the server where the client software package is located as user **user_client**.

Step 6 Decompress the package.

Go to the directory where the package is stored, for example, **/opt/Bigdata/client**. Run the following command to decompress the package to a local directory:

tar -xvf FusionInsight_Cluster_1_Services_Client.tar

Step 7 Verify the software package.

Run the **sha256sum** to verify the retrieved file, for example,

sha256sum -c FusionInsight_Cluster_1_Services_ClientConfig.tar.sha256

```
FusionInsight_Cluster_1_Services_ClientConfig.tar: OK
```

Step 8 Run the following command to decompress the retrieved file:

tar -xvf FusionInsight_Cluster_1_Services_ClientConfig.tar

Step 9 Configure network connections for the client.

1. Ensure that the host where the client is installed can communicate with the hosts listed in the **hosts** file stored in the directory containing the decompressed package, for example, **/opt/Bigdata/client/FusionInsight_Cluster_***<Cluster ID>*_**Services_ClientConfig/hosts**.
2. If the host where the client is installed is not a host in the cluster, you need to set the mapping between the host name and the service plane IP address for each cluster node in **/etc/hosts**, user **root** rights are required to modify the file. Each host name uniquely maps an IP address. You can perform the following steps to import the domain name mapping of the cluster to the **hosts** file:
   a. Switch to the **root** user or a user who has permission to modify the **hosts** file.

      su - root

   b. Go to the directory where the client package is decompressed.

      cd /opt/Bigdata/client/FusionInsight_Cluster_1_Services_ClientConfig

   c. Run the **cat realm.ini >> /etc/hosts** command to import the domain name mapping to the hosts file.

NOTE

- If the host where the client is installed is not a host in the cluster, configure network connections for the client to prevent errors from occurring when you run commands on the client.
- If the Spark task is run in yarn-client mode, add the **spark.driver.host** parameter in the *Client installation directory*/**Spark/spark/conf/spark-defaults.conf** file and set the parameter value to the IP address of the client.
- When yarn-client mode is used, to ensure that the Spark WebUI can properly display information, add the mappings between the client IP addresses and host names to the hosts file on the active and standby Yarn nodes, that is, the ResourceManager nodes in the cluster.

Step 10 Go to the directory where the installation package is stored, and run the following command to install the client to the specified directory (an absolute path), for example, **/opt/hadoopclient**:

cd /opt/Bigdata/client/FusionInsight_Cluster_1_Services_ClientConfig

Run the **./install.sh /opt/hadoopclient** command and wait for the client installation to complete. The client is successfully installed if information similar to the following is displayed:

```
The component client is installed successfully
```

NOTE

- If the **/opt/hadoopclient** directory has been used by the client of all or some installed services, use another directory when another client is installed.
- Delete the client installation directory to uninstall the client.
- To ensure that the client you install can only be used by you, add the **-o** parameter. That is, run the **./install.sh /opt/hadoopclient -o** command to install the client.
- If the NTP server is to be installed in **chrony** mode, ensure that the parameter **chrony** is added in the installation process, that is, run the command **./install.sh /opt/hadoopclient -o chrony** to install the client.
- Because HBase uses the Ruby syntax, if the installed client contains the HBase client, it is recommended that the client installation directory contain only uppercase letters, lowercase letters, digits, and _**-?.@+=** characters.
- If the client node is a server outside the cluster and cannot communicate with the service plane IP address of the active OMS node or cannot access port 20029 of the active OMS node, the client can be successfully installed but cannot be registered with the cluster and cannot be displayed on the GUI.

Step 11 Log in to the client to check whether the client is successfully installed.

1. Run the **cd /opt/hadoopclient** command to go to the client installation directory.
2. Run the **source bigdata_env** command to configure the environment variables for the client.
3. If the cluster is in security mode, run the following command to set kinit authentication and enter the password for logging in to the client, In normal mode, user authentication is not required:

   kinit admin

   ```
   Password for admin@HADOOP.COM:          #Enter the login password of user admin (this password is the same as the user password for cluster login).
   ```

4. Run the **klist** command to view and confirm authentication details.

   ```
   Ticket cache: FILE:/tmp/krb5cc_0
   Default principal: admin@HADOOP.COM

   Valid starting     Expires            Service principal
   04/09/2021 18:22:35  04/10/2021 18:22:29  krbtgt/HADOOP.COM@HADOOP.COM
   ```

   NOTE

   - When kinit authentication is used, the ticket is stored in the **/tmp/krb5cc**_*uid* directory by default.

     uid indicates the ID of the user who logs in to the operating system. For example, if the UID of user **root** is **0**, the ticket generated for kinit authentication after user **root** logs in to the system is stored in the **/tmp/krb5cc_0** directory.

     If the current user does not have the read and write permission on the **/tmp** directory, the ticket generated cache path is changed to *client installation directory*/**tmp/krb5cc_uid**. For example, the client installation directory is **/opt/hadoopclient**. The kinit authentication ticket is stored in **/opt/hadoopclient/tmp/krb5cc_uid**.
   - If kinit authentication is used and the same user is used to log in to the operating system, there is a risk that tickets are overwritten. You can set the **-c cache_name** parameter to specify the ticket buffer location or set the **KRB5CCNAME** environment variable to avoid this problem.

Step 12 After the cluster is reinstalled, the client that has been installed is no longer available. Perform the following operations to reinstall the client.

1. Log in to the node where the client is located as user **root**.
2. Run the following command to check the directory where the client is located. (In the following example, the client is located in the **/opt/hadoopclient** directory.)

   ll /opt

   ```
   drwxr-x---. 6 root root      4096 Dec 11 19:00 hadoopclient
   drwxr-xr-x. 3 root root      4096 Dec  9 02:04 godi
   drwx------. 2 root root     16384 Nov  6 01:03 lost+found
   drwxr-xr-x. 2 root root      4096 Nov  7 09:49 rh
   ```

3. Run the **mv** command to remove the directory where the client program is located and all files in this directory. (For example, remove the **/opt/hadoopclient** directory and all files in it.)

   mv /opt/hadoopclient */tmp/clientbackup*

4. Reinstall the client.

----End

10.9.1.2 Using a Client

Scenario

After a client is installed, you can use shell commands on the client in an O&M scenario or service scenario, or use example projects on the client in an application development scenario.

Use a client in an O&M scenario or service scenario.

Prerequisites

- You have installed the client. For example, the installation directory is **/opt/Bigdata/client**.
- The component service user can be created by the system administrator as required.

  The keytab file must be downloaded for a **Machine-Machine** user. Change the password of the **Human-Machine** user at the first login.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/Bigdata/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster uses the Security Mode, run the following command to perform user authentication. If the cluster uses the Normal Mode, no user authentication is required.

kinit *component service user*

Step 5 Run shell commands based on the actual service requirements.

----End

10.9.1.3 Updating the Configuration of the Installed Client

Scenario

The cluster provides a client for you to connect to a server, view task results, or manage data. If you modify service configuration parameters on FusionInsight Manager and restart the service, you need to download and install the installed client again or use the configuration file to update the client.

Prerequisites

You have installed the client.

Procedure

Method 1:

Step 1 Log in to FusionInsight Manager. Click the wanted cluster from the **Cluster** drop-down list..

Step 2 Choose **More** > **Download Client** > **Configuration Files Only**.

The generated compressed file contains the configuration files of all services.

Step 3 Determine whether a configuration file needs to be generated on the cluster node.

- If yes, select **Save to Path**, and click **OK** to generate the client file. By default, the client file is generated in **/tmp/FusionInsight-Client** on the active management node. You can also store the client file in other directories, and user **omm** has the read, write, and execute permissions on the directories. Then, go to **Step 4**.
- If no, click **OK**, specify a local save path, and download the complete client. Wait until the download is complete, and go to **Step 4**.

Step 4 Use WinSCP to save the compressed file to the installation directory of the client as the client installation user, such as **/opt/hadoopclient**.

Step 5 Decompress the software package.

Run the following commands to enter into the directory where the client is installed, and decompress the file to a local directory. For example, the downloaded client file is **FusionInsight_Cluster_1_Services_Client.tar**.

cd /opt/hadoopclient

tar -xvf FusionInsight_Cluster_1_Services_Client.tar

Step 6 Verify the software package.

Run the **sha256sum** command to verify the retrieved file. Check whether the command output is consistent with the information in the **sha256** file. Example command:

sha256sum -c FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar.sha256

```
FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar: OK
```

Step 7 Decompress the package to obtain the configuration file.

tar -xvf FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar

Step 8 Run the following command in the client installation directory to update the client using the configuration file:

sh refreshConfig.sh *Client installation directory Directory where the configuration file is located*

For example, run the following command:

sh refreshConfig.sh /opt/hadoopclient /opt/hadoopclient/FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

----End

Method 2:

Step 1 Log in to the client installation node as user **root**.

Step 2 Go to the client installation directory, for example, **/opt/Bigdata/client** and run the following command to update the configuration file:

cd /opt/Bigdata/client

sh autoRefreshConfig.sh

Step 3 Enter the username and password of the FusionInsight Manager administrator and the floating IP address of FusionInsight Manager.

Step 4 Enter the names of the components whose configuration needs to be updated. Use commas (,) to separate the component names. Press **Enter** to update the configurations of all components if necessary.

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

----End

10.9.2 Managing Mutual Trust Relationships Between Managers

10.9.2.1 Introduction to Mutual Trust Relationships Between Clusters

Function Description

By default, users of big data clusters in safe mode can only access resources in the cluster. In other clusters, they cannot perform identity authentication to access resources in safe mode.

Features

- **Domain**

  The usage range of users in each system is called a **domain**. Each Manager system must have a unique domain name. Cross-Manager access means users to be used across domains.

- **User Encryption**

  Cross-Manager mutual trust relationships can be configured by using FusionInsight Manager. The current Kerberos server supports only **aes256-cts-hmac-sha1-96:normal** and **aes128-cts-hmac-sha1-96:normal**. Encryption types for encrypting cross-domain users cannot be changed.

- **User Authentication**

  After cross-manager mutual trust is configured, if a user with the same name exists in the two systems and the user with the same name in the peer system has the permission to access a resource in the system, the current system user can access the remote resource.

- **Direct Mutual Trust**

  When cross-cluster mutual trust relationships are built between two clusters, the system saves the mutual-trust receipts. Users can access the remote system through the mutual-trust receipts.

10.9.2.2 Changing Manager System Domain Name

Scenario

The usage range of users in each FusionInsight Manager is called a domain. Each system must have a unique domain name. The domain name of the FusionInsight Manager is generated during installation. To change the domain name to a specific domain name, run the FusionInsight Manager command.

NOTICE

- Changing the system domain name is a high-risk operation. Before performing operations in this section, ensure that the OMS data has been backed up by referring to **Backing Up OMS Data**.

Impact on the System

- During the configuration, all of the clusters need to be restarted and are unavailable during restart.

- During the configuration, the domain names will be changed, and the passwords of Kerberos administrator and OMS Kerberos administrator will be initialized. You need to use the default passwords and change the password. If a component running user whose password is generated randomly by the system is used for identity authentication, see **Exporting an Authentication Credential File** to download the keytab file again.
- After the system domain name is changed, passwords of the **admin** user, component running user, and the **Human-machine** user added by the system administrator before the domain name is changed will be reset to the same. Change the passwords. The reset password consists of two parts: one part is generated by the system and the other is set by the user. The system generating part is Admin@123, which is the default password. For details about the user-defined part, see descriptions of **Password Suffix** in **Table 10-53**. For example, if the system generates **Admin@123** and the user sets **Test#$%@123**, the final initial password is **Admin@123Test#$%@123**.
- The new password must meet the password policies. To obtain the new **Human-machine** user password, log in to the active OMS as user **omm** and run the following script:

  sh ${BIGDATA_HOME}/om-server/om/sbin/get_reset_pwd.sh *passwd_suffix user_name*

  – *passwd_suffix* indicates the user setting part (**Admin@123** by default).

  – *user_name* is optional (**admin** by default).

  Example:

  sh ${BIGDATA_HOME}/om-server/om/sbin/get_reset_pwd.sh Test#$%@123

  ```
  To get the reset password after changing cluster domain name.
    pwd_min_len      : 8
    pwd_char_types   : 4
  The password reset after changing cluster domain name is: "Admin@123Test#$%@123"
  ```

  In this example, **pwd_min_len** and **pwd_char_types** indicate the minimum password length and number of password character types respectively defined in the password policies. **Admin@123Test#$%@123** indicates the **Human-Machine** user password after the system domain name is changed.
- The reset password after the inter-system domain name is changed consists of two parts: one part is generated by the system and the other is set by the user. The reset password must meet the password policy. In case of insufficient length, add one or multiple at-signs (@) between Admin@123 and user setting part. If there are five character types, add a space on the right of Admin@123.

  When the user setting part is **Test@123** and default user password policy is used, the new password is **Admin@123Test@123**. The password contains 17 characters of four types. If the current password policy must be met, process the password according to **Table 10-52**.

Table 10-52 Password processing

Minimum Password Length	Number of Character Types	Password Policy Satisfaction	New Password
8 to 17 characters	4	Password policies are met.	Admin@123Test@123
18 characters	4	Add an at sign (@).	Admin@123@Test@123
19 characters	4	Add two at signs (@).	Admin@123@@Test@123
8 to 18 characters	5	Add a space.	Admin@123 Test@123
19 characters	5	Add a space and an at sign (@).	Admin@123 @Test@123
20 characters	5	Add a space and two at signs (@).	Admin@123 @@Test@123

- After the system domain name is changed, download the **keytab** file for the **Machine-Machine** user added by the system administrator before the domain name is changed.
- After the system domain name is changed, download and install the client again.

Prerequisites

- The system administrator has specified service requirements and planned domain names for the systems.

  A domain name can contain only uppercase letters, digits, dots (.), and underscores (_), and must start with a letter or a digit.
- **Running Status** of all services in the Manager clusters is **Normal**.
- The **acl.compare.shortName** parameter of the ZooKeeper service of all clusters in the Manager must be set to the default value **true**. Otherwise, change the value to **true** and restart the ZooKeeper service.

Procedure

Step 1 Log in to FusionInsight Manager of a cluster.

Step 2 Choose **System** > **Permission** > **Domain and Mutual Trust**.

Step 3 Change parameters.

Table 10-53 Related Parameters

Parameter	Description
Local Domain	Set the value to the domain name of the system.

Parameter	Description
Password Suffix	The user sets part of the **Human-Machine** user after password reset. The default value is **Admin@123**. **NOTE** This parameter is only changed **Local Domain** parameters to take effect. The following conditions must be met: • The password ranges from 8 to 16 characters. • The password must contain at least three types of the following: uppercase letters, lowercase letters, numbers, and the following special characters: `~!@#$%^&*()-_=+|[{}];:',<.>/? and space.

Step 4 Click **OK**. After the modification is complete, proceed with the subsequent steps. Do not perform the subsequent steps in advance.

Step 5 Log in to the active management node as user **omm**.

Step 6 Run the following command to restart the domain update configuration:

sh ${BIGDATA_HOME}/om-server/om/sbin/restart-RealmConfig.sh

The command is run successfully if the following information is displayed:

```
Modify realm successfully. Use the new password to log into FusionInsight again.
```

NOTE

After restart, some hosts and services cannot be accessed and an alarm is generated. This problem can be automatically resolved in about 1 minute after **restart-RealmConfig.sh** is run.

Step 7 Log in to the FusionInsight Manager as the reset user **admin** and password (for example, Admin@123Admin@123). On the FusionInsight Manager home page, click ... next to the name of the cluster to be operated and select **Restart**.

In the displayed confirmation dialog box, click **OK**.

In the displayed dialog box, click **OK**. Wait for a while until a message indicating that the operation is successful is displayed. Click **Finish**.

Step 8 Log out from FusionInsight Manager and then log in to it again. If the login is successful, the configuration is successful.

Step 9 Log in to the active management node as user **omm** and run the following command to update the job submission client configuration:

sh /opt/executor/bin/refresh-client-config.sh

----End

10.9.2.3 Configuring Cross-Manager Cluster Mutual Trust Relationships

Scenario

When two clusters in different security modes need to access each other's resources, the administrator can set up a mutual trust system so that users of external systems can use the system.

The usage range of users in each system is called a **domain**. Each Manager system must have a unique domain name. Cross-Manager access means users to be used across domains.

NOTE

A maximum of 500 mutual trust clusters can be configured.

Impact on the System

- After cross-Cluster mutual trust is configured, users of an external system can be used in the local system. The system administrator needs to periodically check the user rights in the Manager system based on enterprise service and security requirements.
- When configuring cross-cluster mutual trust, you need to stop all clusters, which interrupts services.
- After cross-Cluster mutual trust is configured, each of the clusters trusting each other can add Kerberos internal users "krbtgt/*local cluster domain name*@*external cluster domain name*" and "krbtgt/*external cluster domain name*@*local cluster domain name*". The two users cannot be deleted. Based on enterprise service and security requirements, the system administrator needs to change the password periodically. The passwords of the four users in the two systems trusting each other must be the same. For details, see **Changing the Password for a Component Running User**. Connections of cross-Manager service applications may be affected during the password change.
- After configuring the cross-Cluster mutual trust relationship, download and install the client again for each cluster.
- After cross-Cluster mutual trust is configured, verify services. For information about how to access the resources in the remote system by using users in the local system, see **Assigning User Permissions After Cross-Cluster Mutual Trust Is Configured**.

Prerequisites

- The system administrator has specified service requirements and planned domain names for the systems. A domain name can contain only uppercase letters, digits, dots (.), and underscores (_), and must start with a letter or a digit.
- Before configuring cross-Cluster mutual trust, ensure that the domain names of the two Manager systems are different. When an ECS or BMS cluster is created on MRS, a unique system domain name is randomly generated. Generally, you do not need to change the system domain name.
- Before cross-Cluster mutual trust is configured, ensure that the two systems do not have the same host name or the same IP address.
- Time of two systems configured trust relationships must be consistent and the Network Time Protocol (NTP) service in the two systems must use the same time source.
- **Running Status** of all services in the Manager clusters is **Normal**.
- The **acl.compare.shortName** parameter of the ZooKeeper service of all clusters in the Manager must be set to the default value **true**. Otherwise, change the value to **true** and restart the ZooKeeper service.

Procedure

Step 1 Log in to the FusionInsight Manager of one of the two systems to be configured with mutual trust.

Click ··· > **Stop** next to the cluster to be operated. Enter the administrator password. In the **Stop Cluster** dialog box that is displayed, click **OK**. Wait until the cluster is stopped.

Step 2 Stop all clusters on the home page.

Step 3 Choose **System** > **Permission** > **Domain and Mutual Trust**.

Step 4 Change the **Peer Mutual Trust Domain** parameter

Table 10-54 Related Parameters

Parameters	Description
realm_name	Set the value to the domain name of the peer system.
ip_port	Set the value to the KDC address of the peer system. The parameter value format is *IP address of the node where the Kerberos service of the mutual trust cluster is to be configured in the peer system:port*. • In dual-plane networking, you need to enter the service IP address. • If an IPv6 address is used, the IP address must be enclosed in square brackets (**[]**). • Use a comma to separate the KDC addresses of the active and standby Kerberos services or multiple clusters in the peer system need to establish mutual trust with the local system. • You can obtain the port number by viewing the **kdc_ports** parameter of the KrbServer service. The default value is 21732. You can obtain the IP address of the node where the service is deployed by clicking the **Instances** tab on the **KrbServer** service page and viewing the **Service IP Address** of the **KerberosServer** role. For example, if the Kerberos service is deployed on 10.0.0.1 and 10.0.0.2, to establish mutual trust with the local system, the corresponding parameter value is **10.0.0.1:21732,10.0.0.2:21732**.

NOTE

If you need to configure trust relationships for multiple Manager systems, click + to add a new project and set parameters. A maximum of 16 systems can be mutually trusted. Click — to delete redundant configurations.

Step 5 Click **OK**.

Step 6 Log in to the active management node using the active management IP address as user **omm**. Run the following command to update domain configuration:

sh ${BIGDATA_HOME}/om-server/om/sbin/restart-RealmConfig.sh

The command is run successfully if the following information is displayed:

```
Modify realm successfully. Use the new password to log into FusionInsight again.
```

After restart, some hosts and services cannot be accessed and an alarm is generated. This problem can be automatically resolved in about 1 minute after **restart-RealmConfig.sh** is run.

Step 7 Log in to the FusionInsight Manager and start all clusters.

Click ⋯ > **Start** next to the cluster to be operated. In the **Start Cluster** dialog box that is displayed, click **OK**. Wait until the cluster is started.

Step 8 Log in to FusionInsight Manager of the other system and repeat the preceding operations.

----End

10.9.2.4 Assigning User Permissions After Cross-Cluster Mutual Trust Is Configured

Scenario

After cross-Cluster mutual trust is configured, assign users the access permission for the systems so that the users can perform required service operations in the systems.

Prerequisites

The mutual trust relationship between Manager systems has been configured.

Procedure

Step 1 Log in to FusionInsight Manager of local system.

Step 2 Choose **System** > **Permission** > **User** to check whether the user who performs the service operation exists.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Click ⌄ on the left of a specified user, check whether the permissions assigned for the user group where the user resides and the role meet the service requirements. If not, create a role and bind the role to the user, or modify the permissions of the user group or role. For details, see **Configuring Permissions**.

Step 4 Create the user required by the service and associate the required user group or role. For details, see **Creating a User**.

Step 5 Log in to FusionInsight Manager of remote system and repeat **Step 2** to **Step 4** to create the same user as that in cluster A and assign required permissions.

----End

10.9.3 Configuring Periodical Alarm and Audit Information Backup

Scenario

Modify the related configuration file to periodically back up the alarm, audit information of FusionInsight Manager and service audit information of to the specified storage location.

The backup can be performed using FTP or SFTP. FTP does not encrypt data, which may cause potential security risks. Therefore, SFTP is recommended.

Procedure

Step 1 Log in to the active management node using the active management IP address as user **omm**.

NOTE

Perform this operation only on the active management node. This operation is not supported on the standby management node.

Step 2 Run the following command to go to the related directory:

cd ${BIGDATA_HOME}/om-server/om/sbin

Step 3 Run the following command to configure periodically the alarm, audit information of FusionInsight Manager and service audit information backup:

./setNorthBound.sh -t *Information type* **-i** *Remote server IP address* **-p** *SFTP or FTP port used by the server* **-u** *Username* **-d** *Save path* **-c** *Interval (minute)* **-m** *Number of information records in each file* **-s** *Flag for enabling or disabling backup* **-e** *Specified protocol*

Example:

./setNorthBound.sh -t alarm -i 10.0.0.10 -p 22 -u sftpuser -d /tmp/ -c 10 -m 100 -s true -e sftp

This script will modify the alarm information backup configuration file **alarm_collect_upload.properties**. The file save path is **${BIGDATA_HOME}/om-server/tomcat/webapps/web/WEB-INF/classes/config**.

./setNorthBound.sh -t audit -i 10.0.0.10 -p 22 -u sftpuser -d /tmp/ -c 10 -m 100 -s true -e sftp

This script will modify the audit information backup configuration file **audit_collect_upload.properties**. The file save path is **${BIGDATA_HOME}/om-server/tomcat/webapps/web/WEB-INF/classes/config**.

./setNorthBound.sh -t service_audit -i 10.0.0.10 -p 22 -u sftpuser -d /tmp/ -c 10 -m 100 -s true -e sftp

This script will modify the service audit information backup configuration file **service_audit_collect_upload.properties**. The file save path is **${BIGDATA_HOME}/om-server/tomcat/webapps/web/WEB-INF/classes/config**.

Step 4 Enter the password as prompted. The password is encrypted and saved in the configuration file.

```
Please input sftp/ftp server password:
```

Step 5 If the following information is displayed, the operation is successful. The configuration file will be automatically synchronized to the standby management node.

```
execute command syncfile successfully.
Config Succeed.
```

----End

10.9.4 Modifying the Manager Routing Table

Scenario

After the FusionInsight Manager is installed, the system automatically creates two pieces of routing information on the active management node. Run the **ip rule list** command to view the routing information, as shown in the following example:

```
0:  from all lookup local
32764:  from all to 10.10.100.100 lookup ntp_rt  #FusionInsight Manager NTP routing information created by the system. The information is not displayed when no external NTP clock source is configured.
32765:  from 192.168.0.117 lookup om_rt   #OM route information created on FusionInsight Manager
32766:  from all lookup main
32767:  from all lookup default
```

NOTE

If the external NTP server has not been configured, only the OM routing information **om_rt** will be created.

When the routing information created by the FusionInsight Manager system conflicts with the routing information of the enterprise, use the **autoroute.sh** tool to disable or enable routes created by FusionInsight Manager.

Impact on the System

After the routing information created by the FusionInsight Manager system is disabled and before the new routing information is set, FusionInsight Manager cannot be accessed but operating of the cluster will not be affected.

Prerequisites

The Manager has already been installed.

The information about the WS floating IP route to be created has already obtained.

Disable the routing information created by the system.

Step 1 Log in to the active management node as user **omm**. Run the following commands to disable the routing information created by the system:

cd ${BIGDATA_HOME}/om-server/om/sbin

./autoroute.sh disable

```
Deactivating Route.
Route operation (disable) successful.
```

Step 2 Run the following command to view the execution result:

ip rule list

```
0:  from all lookup local
32766:  from all lookup main
32767:  from all lookup default
```

Step 3 Run the following command and enter the password of user **root** to switch to user **root**:

su - root

Step 4 Run the following commands to manually create *WS floating IP address* routing information:

ip route add *WS floating IP address Network segment number/WS floating IP address subnet mask* **scope link src** *WS floating IP address* **dev** *WS floating IP address NIC* **table om_rt**

ip route add default via *WS floating IP address gateway* **dev** *WS floating IP address NIC* **table om_rt**

ip rule add from *WS floating IP address* **table om_rt**

For example,

ip route add 192.168.0.0/255.255.255.0 scope link src 192.168.0.117 dev eth0:ws table om_rt

ip route add default via 192.168.0.254 dev eth0:ws table om_rt

ip rule add from 192.168.0.117 table om_rt

NOTE

If the IP address mode of the current network is IPv6, run the **ip -6 route add** command.

Step 5 Run the following commands to manually create *NTP service* routing information (Skip this step when no external NTP clock source is configured.):

ip route add default via *NtpIP address gateway* **dev** *NIC of the local host IP address* **table ntp_rt**

ip rule add to *NTP IP address* **table ntp_rt**

NIC of the local host IP address indicates the NIC that can interwork with the network segment where the NTP server is located.

For example,

ip route add default via 10.10.100.254 dev eth0 table ntp_rt

ip rule add to 10.10.100.100 table ntp_rt

Step 6 Run the following command to view the execution result. For example, if routing information of which the routing table names are **om_rt** and **ntp_rt** is generated, the operation is successful.

ip rule list

```
0:  from all lookup local
32764:  from all to 10.10.100.100 lookup ntp_rt #The information is not displayed when no external NTP clock source is configured.
32765:  from 192.168.0.117 lookup om_rt
32766:  from all lookup main
32767:  from all lookup default
```

----End

Enable the routing information created by the system.

Step 1 Log in to the active management node as user **omm**.

Step 2 Run the following commands to enable the routing information created by the system:

cd ${BIGDATA_HOME}/om-server/om/sbin

./autoroute.sh enable

```
Activating Route.
Route operation (enable) successful.
```

Step 3 Run the following command to view the execution result. For example, if routing information of which the routing table names are **ntp_rt** and **om_rt** is generated, the operation is successful.

ip rule list

```
0:  from all lookup local
32764:  from all to 10.10.100.100 lookup ntp_rt  #The information is not displayed when no external NTP clock source is configured.
32765:  from 192.168.0.117 lookup om_rt
32766:  from all lookup main
32767:  from all lookup default
```

----End

10.9.5 Switching to Maintenance Mode

Scenario

FusionInsight Manager allows you to set clusters, services, hosts, or OMSs to the maintenance mode. In this way, objects in the maintenance mode do not report alarms. This prevents the system from generating a large number of meaningless alarms during maintenance and changes such as upgrade, which affects O&M personnel's judgment on the cluster status.

- Cluster maintenance mode

  If the cluster is not officially brought online or is temporarily offline for O&M operations (for example, non-rolling upgrade), you can set the entire cluster to the maintenance mode.

- Service maintenance mode

  When maintaining a specific service (for example, performing commissioning operations that may affect services, such as restarting service instances in batches, powering on or off nodes related to the service, or repairing the service), you can configure only the involved services to the maintenance mode.

- Host maintenance mode

  When performing maintenance operations on a host (for example, powering on or off a node, isolating a host, reinstalling a host, upgrading the OS, or replacing nodes), you can set the involved hosts to the maintenance mode.

- OMS maintenance mode

  You can set the OMS node to the maintenance mode when restarting, replacing, or repairing the OMS node.

Impact on the System

In maintenance mode, the alarms that are not caused by maintenance operations are also suppressed. The alarms can be reported only after the system exits the maintenance mode. Therefore, exercise caution when setting the maintenance mode.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Set the maintenance mode.

Determine the object for which maintenance mode needs to be configured based on the actual operation scenario. For details, see **Table 10-55**.

Table 10-55 Switching to maintenance mode

Scenario	Procedure
Configuring a cluster to enter maintenance mode	1. On the home page, click ··· next to the target cluster name and click **Enter Maintenance Mode**. 2. In the dialog box that is displayed, click **OK**. After the cluster enters maintenance mode, the status of the cluster is displayed as . After maintenance is complete, click **Exit Maintenance Mode**. The cluster exits maintenance mode.
Configuring services to enter maintenance mode	1. On the management page, choose **Cluster** > *Name of the desired cluster* > **Services** > *Service name*. 2. On the service details page, choose **More** > **Enter Maintenance Mode**. 3. In the dialog box that is displayed, click **OK**. After a service enters maintenance mode, the status of the service in the service list is displayed as . After maintenance is complete, click **Exit Maintenance Mode**. The service exits the maintenance mode. **NOTE** When configuring a service to enter the maintenance mode, you are advised to set other upper-layer services that depend on the service to the maintenance mode.

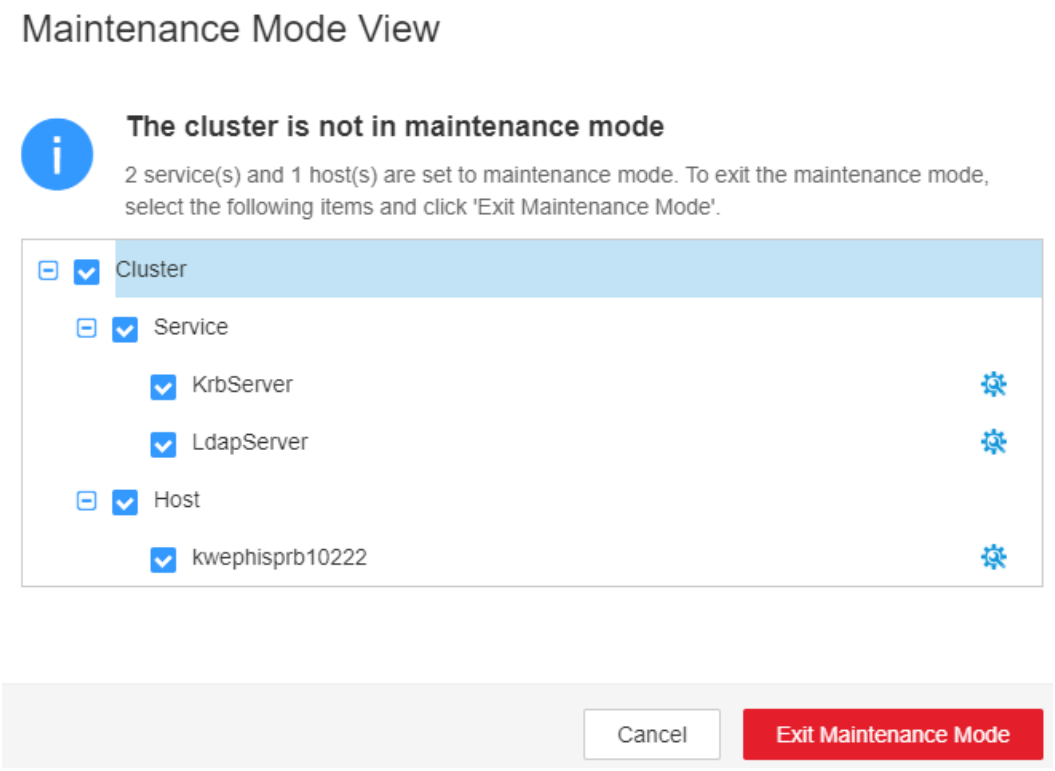
Scenario	Procedure
Configuring a host to enter maintenance mode	1. Click **Hosts** on the management page. 2. On the host page, select the target host and choose **More** > **Enter Maintenance Mode**. 3. In the dialog box that is displayed, click **OK**. After the host enters maintenance mode, the status of the host in the host list is displayed as  . After maintenance is complete, choose **More** > **Exit Maintenance Mode**. The host exits maintenance mode.
Configuring the OMS to enter maintenance mode	1. On the management page, choose **System** > **OMS** > **Enter Maintenance Mode**. 2. In the dialog box that is displayed, click **OK**. After the OMS enters maintenance state, the OMS status is displayed as  . After maintenance is complete, click **Exit Maintenance Mode**. The OMS exits maintenance mode.

Step 3 View the cluster maintenance view.

On the home page of the management page, click ··· next to the name of the target cluster and click **O&M View**. In the displayed window, view the services and hosts in maintenance mode in the cluster.

After the maintenance is complete, you can select services and hosts in batches in the maintenance mode view and click **Exit Maintenance Mode** to exit the maintenance mode in batches.

Figure 10-19 Exiting maintenance mode in batches



----End

10.9.6 Routine Maintenance

To ensure a long-term proper and stable running of the system, system administrators or maintenance engineers need to check the items listed in **Table 10-56** periodically and rectify faults based on the check results. It is recommended that system administrators record the result in each task scenario and sign off based on the enterprise management regulations.

Table 10-56 Check items

Routine Maintenance Period	Task	Routine Maintenance Content
Every day	Checking the cluster service status	• Check whether the running status, and configuration status of each service are normal and whether the status icons are in green. • Check whether the running status, and configuration status of the role instances of each service are normal and whether the status icons are in green. • Check whether the active/standby status of role instances of each service can be properly displayed. • Check whether the **Dashboard** results of services and role instances are normal.
	Checking the cluster host status	• Check whether the running status of each host is normal and whether the status icon is in green. • Check the current disk usage, memory usage, and CPU usage of each host. Check whether the current memory usage and CPU usage are ascending.
	Checking the cluster alarm information	Check whether there are alarms generated in the previous day and automatically cleared.
	Checking the cluster audit information	Check whether there are **Critical** and **Major** operations performed in the previous day and whether the operations are valid.
	Checking the cluster backup	Check whether the OMS, LDAP, DBService, and NameNodeOMS, LDAP, and DBService were automatically backed up in the previous day.
	Checking the health check results	Perform the health check on FusionInsight Manager, and download the health check report to check whether any exception exists in the current cluster. You are advised to enable the automatic health check, export the latest cluster health check result, and repair unhealthy items based on the result.

Routine Maintenance Period	Task	Routine Maintenance Content
	Checking the network communication	Check the cluster network running status and check whether delay exists in the network communication between nodes.
	Checking the storage status	Check whether the total amount of cluster data storage increases suddenly. ● Check whether the disk usage is reaching the threshold, and find the causes, such as whether there is junk data or cold data needs to be deleted. ● Check whether the services are increasing and whether the disk partitions need to be expanded.
	Checking logs	● Check whether any failed or suspended MapReduce or Spark job exists, view the **/tmp/logs/${username}/logs/${application id}** log file in HDFS, and rectify the fault. ● Check the Yarn job logs, view the logs recording failed or suspended jobs, and delete the duplicate data. ● Check the worker logs of Storm. ● Back up logs to the storage server.
Every week	Managing users	Check whether the user passwords are about to expire and notify users to change their passwords. To change the password of a **Machine-Machine** user, the keytab file needs to be downloaded again.
	Analyzing alarms	Export the alarms generated in a specified period and analyze them.
	Scanning disks	Check the disk health status. You are advised to use professional disk health check tools to perform the check.
	Collecting statistics of storage	Check the cluster node disk data in batches and check whether the data is evenly stored. Select the disks where the data amount is too large or too small and check whether the disks are normal.
	Recording changes	Arrange and record the operations on cluster configuration parameters and files to provide references for fault analysis and rectification.

Routine Maintenance Period	Task	Routine Maintenance Content
Every month	Analyzing logs	• Collect and analyze the hardware logs of cluster node servers, such as the BMC system logs. • Collect and analyze the OS logs of cluster node servers. • Collect and analyze the cluster logs.
	Diagnosing the network	Analyze the cluster network health status.
	Managing hardware	Check the equipment rooms where the devices are running and clean the devices.

10.10 Log Management

10.10.1 About Logs

Log Description

The log file path of the MRS cluster is **/var/log/Bigdata**. The following table lists the log types.

Table 10-57 Log types

Log Type	Description
Installation logs	Installation logs record information about FusionInsight Manager, cluster, and service installation to help users locate installation errors.
Run logs	Run logs record the running track information, debugging information, status changes, potential problems, and error information generated during the running of services.
Audit logs	Audit logs record information about users' activities and operation instructions, which can be used to locate fault causes in security events and determine who are responsible for these faults.

The following table lists the MRS log directories.

Table 10-58 Log directories

Directory	Log
/var/log/Bigdata/audit	Component audit log.
/var/log/Bigdata/controller	Log collecting script log. Controller process log. Controller monitoring log.
/var/log/Bigdata/dbservice	DBService log.
/var/log/Bigdata/flume	Flume log.
/var/log/Bigdata/hbase	HBase log.
/var/log/Bigdata/hdfs	HDFS log.
/var/log/Bigdata/hive	Hive log.
/var/log/Bigdata/httpd	httpd log.
/var/log/Bigdata/hue	Hue log.
/var/log/Bigdata/kerberos	Kerberos log.
/var/log/Bigdata/ldapclient	LDAP client log.
/var/log/Bigdata/ldapserver	LDAP server log.
/var/log/Bigdata/loader	Loader log.
/var/log/Bigdata/logman	logman script log management log.
/var/log/Bigdata/mapreduce	MapReduce log.
/var/log/Bigdata/nodeagent	NodeAgent log.
/var/log/Bigdata/okerberos	OMS Kerberos log.
/var/log/Bigdata/oldapserver	OMS LDAP log.
/var/log/Bigdata/metric_agent	Run log file of MetricAgent.
/var/log/Bigdata/omm	oms: complex event processing log, alarm service log, HA log, authentication and authorization management log, and monitoring service run log of the omm server. oma: installation log and run log of the omm agent. core: dump log generated when the omm agent and the HA process are suspended.
/var/log/Bigdata/spark2x	Spark2x log.
/var/log/Bigdata/sudo	Log generated when the **sudo** command is executed by user **omm**.

Directory	Log
/var/log/Bigdata/timestamp	Time synchronization management log.
/var/log/Bigdata/tomcat	Tomcat log.
/var/log/Bigdata/watchdog	Watchdog log.
/var/log/Bigdata/yarn	YARN log.
/var/log/Bigdata/zookeeper	ZooKeeper log.
/var/log/Bigdata/oozie	Oozie log.
/var/log/Bigdata/kafka	Kafka log.
/var/log/Bigdata/storm	Storm log.
/var/log/Bigdata/upgrade	OMS upgrade log file.
/var/log/Bigdata/update-service	Upgrade service logs.

NOTE

After the multi-instance function is enabled, if the system administrator adds multiple HBase, Hive, and Spark service instances, the log description, log level, and log format of the newly added service instances are the same as those of the original service logs. Service instance logs are stored separately in the **/var/log/Bigdata/***servicenameN* directory. The audit logs of the HBase and Hive service instances are stored in the **/var/log/Bigdata/audit/***servicenameN* directory. For example, the logs of HBase1 are stored in the **/var/log/Bigdata/hbase1** and **/var/log/Bigdata/audit/hbase1** directories.

Installation Logs

Table 10-59 Installation logs

Installation Log	Description
Configuration log	Records information about the configuration process before the installation.
FusionInsight Manager installation log	Records information about the two-node FusionInsight Manager installation.
Cluster installation log	Records information about the cluster installation.

Run Logs

Table 10-60 describes the running information recorded in run logs.

Table 10-60 Running information

Run Log	Description
Installation preparation log	Records information about preparations for the installation, such as the detection, configuration, and feedback operation information.
Process startup log	Records information about the commands executed during the process startup.
Process startup exception log	Records information about exceptions during process startup, such as dependent service errors and insufficient resources.
Process run log	Records information about the process running track information and debugging information, such as function entries and exits as well as cross-module interface messages.
Process running exception log	Records errors that cause process running errors, for example, the empty input objects or encoding or decoding failure.
Process running environment log	Records information about the process running environment, such as resource status and environment variables.
Script log	Records information about the script execution process.
Resource reclamation log	Records information about the resource reclaiming process.
Uninstallation clearing logs	Records information about operations performed during service uninstallation, such as directory deletion and execution time

Audit Logs

Audit information recorded in audit logs includes FusionInsight Manager audit information and component audit information.

Table 10-61 Audit information of FusionInsight Manager

Operation Type	Operation
User management	Creating a user. Modifying a user. Deleting a user. Creating a user group. Modifying a user group. Deleting a group. Adding a role. Changing the user's roles. Deleting a role. Changing a password policy. Changing a password. Resetting a password. Logging in. Logging out. Unlocking the screen. Downloading the authentication credential. Unauthorized operation. Unlocking a user account. Locking a user account. Locking the screen. Exporting a user. Exporting a user group. Exporting a role.

Operation Type	Operation
Cluster	Starting a cluster. Stopping a cluster. Restarting a cluster. Performing a rolling restart of a cluster. Restarting all expired instances. Saving configurations. Synchronizing cluster configurations. Customizing cluster monitoring indicators. Configuring monitoring dump. Saving monitoring thresholds. Downloading a client configuration file. Configuring the northbound Syslog interface. Configuring the northbound SNMP interface. Clearing alarms using SNMP. Adding a trap target using SNMP. Deleting a trap target using SNMP. Checking alarms using SNMP. Synchronizing alarms using SNMP. Creating a threshold template. Deleting a threshold template. Applying a threshold template. Saving cluster monitoring configurations. Exporting configurations. Importing cluster configurations. Exporting an installation template. Modifying a threshold template. Canceling the application of a threshold template. Masking an alarm. Sending an alarm. Changing the OMS database password. Resetting the component database password. Restarting OMM and Controller. Starting the health check of a cluster. Importing a certificate file. Configuring SSO information. Deleting historical health check reports. Modifying cluster properties.

Operation Type	Operation
	Running maintenance commands in synchronous mode. Running maintenance commands in asynchronous mode. Customizing report monitoring indicators. Exporting report monitoring data. Runing a command in asynchronous mode using SNMP. Restarting the web service. Customizing monitoring indicators for static resource pools. Exporting monitoring data of a static resource pool. Customizing dashboard monitoring indicators. Stopping a task. Restoring configurations. Modifying domain and trust relationship configurations. Modifying system parameters. Making a cluster enter the maintenance mode. Making a cluster exit the maintenance mode. Making OMS enter the maintenance mode. Making OMS exit the maintenance mode. Making services in a cluster exit the maintenance mode in batches. Modifying OMS configurations. Enabling threshold alarms. Synchronizing all cluster configurations.

Operation Type	Operation
Service	Starting a service. Stopping a service. Synchronizing service configurations. Refreshing a service queue. Customizing service monitoring indicators. Restarting a service. Performing a rolling service restart Exporting service monitoring data. Importing service configuration data. Starting the health check of a service. Configuring a service. Uploading a configuration file. Downloading a configuration file. Synchronizing instance configurations. Commissioning an instance. Decommissioning an instance. Starting an instance. Stopping an instance. Customizing instance monitoring indicators. Restarting an instance. Performing a rolling restart of an instance. Exporting instance monitoring data. Importing instance configuration data. Creating an instance group. Modifying an instance group. Deleting an instance group. Moving an instance to another instance group. Making a service enter the maintenance mode. Making a service exit the maintenance mode. Changing the name of a service. Modifying service association. Downloading monitoring data. Masking alarms. Unmasking alarms. Exporting report data of a service. Adding custom parameters for a report. Modifying custom parameters of a report. Deleting custom parameters of a report.

Operation Type	Operation
	Switching over control nodes. Adding a mount table. Modifying a mount table.
Host	Setting a node rack. Starting all roles. Stopping all roles. Isolating a host. Canceling isolation of a host. Customizing host monitoring indicators. Exporting host monitoring data. Making a host enter the maintenance mode. Making a host exit the maintenance mode. Exporting basic host information. Exporting host distribution report data. Exporting host trend report data. Exporting host cluster report data. Exporting report data of a service. Customizing host cluster monitoring indicators. Customizing host cluster trend monitoring indicators.
Alarm	Exporting alarms. Clearing alarms. Exporting events. Clearing alarms in batches.
Log collection	Collecting log files. Downloading log files. Collecting service stack information. Collecting instance stack information. Preparing service stack information. Preparing instance stack information. Clearing service stack information. Clearing instance stack information.
Audit log	Modifying audit dump configurations. Exporting audit logs.

Operation Type	Operation
Data backup and restoration	Creating a backup task. Executing a backup task. Executing backup tasks in batches. Stopping a backup task. Deleting a backup task. Modifying a backup task. Locking a backup task. Unlocking a backup task. Creating a restoration task. Executing a restoration task. Stopping a restoration task. Retrying a restoration task. Deleting a restoration task.

Operation Type	Operation
Multi-tenant	Saving static configurations. Adding a tenant. Deleting a tenant. Associating a service with a tenant. Deleting a service from a tenant. Configuring resources. Creating a resource. Deleting a resource. Adding a resource pool. Modifying a resource pool. Deleting a resource pool. Restoring tenant data. Modifying global configurations of a tenant. Modifying queue configurations of a capacity scheduler. Modifying queue configurations of a super scheduler. Modifying resource distribution of a capacity scheduler. Clearing resource distribution of a capacity scheduler. Modifying resource distribution of a super scheduler. Clearing resource distribution of a super scheduler. Adding a resource catalog. Modifying a resource catalog. Deleting a resource catalog. Customizing tenant monitoring indicators.

Operation Type	Operation
Health check	Starting the health check of a cluster. Starting the health check of a service. Starting the health check of a host. Starting the health check of OMS. Starting system health check. Updating the health check configuration. Exporting health check reports. Exporting health check results of a cluster. Exporting health check results of a service. Exporting health check results of a host. Deleting historical health check reports. Exporting historical health check reports. Downloading a health check report.

Table 10-62 Component audit information

Audit Log	Operation Type	Operation
ClickHouse audit log	Maintenance management	Granting permissions Revoking permissions Authentication and login information
	Service operation	Creating databases or tables Inserting, deleting, querying, and migrating data
DBService audit log	Maintenance management	Performing backup restoration operations.
HBase audit logs	DDL (data definition) statement	Creating a table. Deleting a table. Modifying a table. Adding a column family. Modifying a column family. Deleting a column family. Enabling a table. Disabling a table. Modifying user information. Changing a password. Logging in.

Audit Log	Operation Type	Operation
	DML (data operation) statement	Putting data (to the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables). Deleting data (from the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables) Checking and importing data (for the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables). Checking and deleting data (the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables).
	Permission control	Assigning permissions users. Canceling user authorization.
HDFS audit logs	Rights management	File/Folder access permission. File/folder owner information.
	File operation	Creating a folder. Creating a file. Opening a file. Appending file content. Changing a file name. Deleting a file or folder. Setting time property of a file. Setting the number of file copies. Merging files. Checking the file system. File link.
Hive audit logs	Metadata operation	Defining metadata, such as creating databases and tables. Deleting metadata, such as deleting databases and tables. Modifying metadata, such as adding columns and renaming tables. Importing and exporting metadata.
	Data maintenance	Loading data to a table. Inserting data into a table.
	Rights management	Creating or deleting a role. Granting/Reclaiming roles. Granting/Reclaiming permissions.
Hue audit log	Service startup	Starting Hue.

Audit Log	Operation Type	Operation
	User operation	User login. User logout.
	Task operations	Creating a task. Modifying a task. Deleting a task. Submitting a task. Saving a task. Updating the status of a task.
KrbServer audit log	Maintenance management	Changing the password of a Kerberos account. Adding a Kerberos account. Deleting a Kerberos account. Authenticating users.
LdapServer audit log	Maintenance management	Adding an operating system user. Adding a user group. Adding a user to user group. Deleting a user. Deleting a group.
Loader audit logs	Security management	User login.
	Metadata management	Querying connector information. Querying a framework. Querying step information.
	Data source connection management	Querying a data source connection. Adding a data source connection. Updating a data source connection. Deleting a data source connection. Activating a data source connection. Disabling a data source connection.

Audit Log	Operation Type	Operation
	Job management	Querying a job. Creating a job. Updating a job. Deleting a job. Activating a job. Disabling a job. Querying all execution records of a job. Querying the latest execution record of a job. Submitting a job. Stopping a job.
MapReduce audit log	Application running	Starting a Container request. Stopping a Container request. After Container request is completed, the status of the request is displayed as succeeded. After Container request is completed, the status of the request is displayed as failed. After Container request is completed, the status of the request is displayed as suspended. Submitting a task. Ending a task.
Oozie audit log	Task management	Submitting a task. Starting a task. Killing a task. Suspending a task. Resuming a task. Running a task again.
Spark2x audit logs	Metadata operations	Defining metadata, such as creating databases and tables. Deleting metadata, such as deleting databases and tables. Modifying metadata, such as adding columns and renaming tables. Importing and exporting metadata.
	Data maintenance	Loading data to a table Inserting data into a table

Audit Log	Operation Type	Operation
Storm audit log	Nimbus	Submitting a topology. Stopping a topology. Reallocating a topology. Deactivating a topology. Activating a topology.
	UI	Stopping a topology. Reallocating a topology. Deactivating a topology. Activating a topology.
Yarn audit logs	Job submission	Submits a job to a queue.
ZooKeeper audit logs	Rights management	Setting access permission to Znode.
	Znode operation	Creating Znodes. Deleting Znodes. Configuring Znode data

FusionInsight Manager audit logs are stored in the database. You can view and export the audit logs on the **Audit** page.

The following table lists the directories to store component audit logs. Audit log files of some components are stored in **/var/log/Bigdata/audit**, such as HDFS, HBase, MapReduce, Hive, Hue, Yarn, Storm, and ZooKeeper. The component audit logs are automatically compressed and backed up to **/var/log/Bigdata/audit/bk** at 03: 00 every day. A maximum of latest 90 compressed backup files are retained, and the backup time cannot be changed. Configure the number of reserved audit log files. For details, see **Configuring the Number of Local Backup Audit Log Files**.

Audit log files of other components are stored in the component log directory.

Table 10-63 Directory for storing component audit logs

Component	Audit Log Directory
DBService	/var/log/Bigdata/audit/dbservice/dbservice_audit.log

Component	Audit Log Directory
HBase	/var/log/Bigdata/audit/hbase/hm/hbase-audit-hmaster.log /var/log/Bigdata/audit/hbase/hm/hbase-ranger-audit-hmaster.log /var/log/Bigdata/audit/hbase/rs/hbase-audit-regionserver.log /var/log/Bigdata/audit/hbase/rs/hbase-ranger-audit-regionserver.log /var/log/Bigdata/audit/hbase/rt/hbase-audit-restserver.log /var/log/Bigdata/audit/hbase/ts/hbase-audit-thriftserver.log
HDFS	/var/log/Bigdata/audit/hdfs/nn/hdfs-audit-namenode.log /var/log/Bigdata/audit/hdfs/nn/ranger-plugin-audit.log /var/log/Bigdata/audit/hdfs/dn/hdfs-audit-datanode.log /var/log/Bigdata/audit/hdfs/jn/hdfs-audit-journalnode.log /var/log/Bigdata/audit/hdfs/zkfc/hdfs-audit-zkfc.log /var/log/Bigdata/audit/hdfs/httpfs/hdfs-audit-httpfs.log /var/log/Bigdata/audit/hdfs/router/hdfs-audit-router.log
Hive	/var/log/Bigdata/audit/hive/hiveserver/hive-audit.log /var/log/Bigdata/audit/hive/hiveserver/hive-rangeraudit.log /var/log/Bigdata/audit/hive/metastore/metastore-audit.log /var/log/Bigdata/audit/hive/webhcat/webhcat-audit.log
Hue	/var/log/Bigdata/audit/hue/hue-audits.log
Kafka	/var/log/Bigdata/audit/kafka/audit.log
Loader	/var/log/Bigdata/loader/audit/default.audit
MapReduce	/var/log/Bigdata/audit/mapreduce/jobhistory/mapred-audit-jobhistory.log
Oozie	/var/log/Bigdata/audit/oozie/oozie-audit.log
Spark2x	/var/log/Bigdata/audit/spark2x/jdbcserver/jdbcserver-audit.log /var/log/Bigdata/audit/spark2x/jdbcserver/ranger-audit.log /var/log/Bigdata/audit/spark2x/jobhistory/jobhistory-audit.log
Storm	/var/log/Bigdata/audit/storm/logviewer/audit.log /var/log/Bigdata/audit/storm/nimbus/audit.log /var/log/Bigdata/audit/storm/supervisor/audit.log /var/log/Bigdata/audit/storm/ui/audit.log
Yarn	/var/log/Bigdata/audit/yarn/rm/yarn-audit-resourcemanager.log /var/log/Bigdata/audit/yarn/rm/ranger-plugin-audit.log /var/log/Bigdata/audit/yarn/nm/yarn-audit-nodemanager.log

Component	Audit Log Directory
ZooKeeper	/var/log/Bigdata/audit/zookeeper/quorumpeer/zk-audit-quorumpeer.log

10.10.2 Manager Log List

Log Description

Log path: The default storage path of Manager log files is **/var/log/Bigdata/***Manager component*.

- ControllerService: **/var/log/Bigdata/controller/** (operation & maintenance system (OMS) installation and run logs)
- Httpd: **/var/log/Bigdata/httpd** (httpd installation and run logs)
- logman: **/var/log/Bigdata/logman** (log packaging tool logs)
- NodeAgent: **/var/log/Bigdata/nodeagent** (NodeAgent installation and run logs)
- okerberos: **/var/log/Bigdata/okerberos** (okerberos installation and run logs)
- oldapserver: **/var/log/Bigdata/oldapserver** (oldapserver installation and run logs)
- MetricAgent: **/var/log/Bigdata/metric_agent** (MetricAgent run log)
- omm: **/var/log/Bigdata/omm** (omm installation and run logs)
- timestamp: **/var/log/Bigdata/timestamp** (NodeAgent startup time logs)
- tomcat: **/var/log/Bigdata/tomcat** (Web process logs)
- watchdog: **/var/log/Bigdata/watchdog** (watchdog logs)
- Upgrade: **/var/log/Bigdata/upgrade** (OMS log upgrade)
- UpdateService: **/var/log/Bigdata/update-service** (upgrade service log)
- Sudo: **/var/log/Bigdata/sudo** (sudo script execution log)
- OS: **/var/log/***message file* (OS system log)
- OS Performance: **/var/log/osperf** (OS performance statistics log)
- OS Statistics: **/var/log/osinfo/statistics** (OS parameter configuration log)

Log archive rule:

The automatic compression and archiving function is enabled for Manager logs. By default, when the size of a log file exceeds 10 MB, the log file is automatically compressed. The naming rule of a compressed log file is as follows: <*Original log name*>-<*yyyy-mm-dd_hh-mm-ss*>.[*ID*].**log.zip** A maximum of 20 latest compressed files are retained.

Table 10-64 Manager logs

Log Type	Log File Name	Description
Controller run log	controller.log	Log file that records component installation, upgrade, configuration, monitoring, alarms, and routine O&M operations
	controller_client.log	Run log file of the Representational State Transfer (REST) API
	acs.log	Acs run log file
	acs_spnego.log	spnego user logs in ACS
	aos.log	Aos run log file
	plugin.log	Aos plug-in logs
	backupplugin.log	Run log file that records the backup and restoration operations
	controller_config.log	Configuration run log file
	controller_nodesetup.log	Controller loading task log file
	controller_root.log	System log file of the Controller process
	controller_trace.log	Log file that records the remote procedure call (RPC) communication between Controller and NodeAgent
	controller_monitor.log	Monitoring logs
	controller_fsm.log	State machine log file
	controller_alarm.log	Controller alarm logs
	controller_backup.log	Controller backup and recovery logs
	install.log, restore_package.log, installPack.log, distributeAdapterFiles.log, install_os_optimization.log	OMS installation log file

Log Type	Log File Name	Description
	oms_ctl.log	OMS startup and stop logs
	preInstall_client.log	Preprocessing logs before client installation
	installntp.log	NTP installation log file
	modify_manager_param.log	Modifying Manager parameter logs
	backup.log	OMS backup script running log file
	supressionAlarm.log	Alarm script run log file
	om.log	OM certificate generation log file
	backupplugin_ctl.log	Startup logs of the backup and restoration plug-in process
	getLogs.log	Run logs of the collection log script
	backupAuditLogs.log	Audit log backup script run log
	certStatus.log	Log file that records regular certificate checks
	distribute.log	Certificate distribution log
	ficertgenetrate.log	Certificate replacement logs, including level-2 certificates, cas certificates, and httpd certificates
	genPwFile.log	Log file that records the generation of certificate password files
	modifyproxyconf.log	Log file for modifying the HTTPD proxy configuration
	importTar.log	Log file that records the process of importing certificates into the trust library.

Log Type	Log File Name	Description
Httpd	install.log	Httpd installation log file
	access_log, error_log	Httpd run log file
logman	logman.log	Log packaging tool log file
NodeAgent	install.log, install_os_optimization.log	NodeAgent installation log file
	installntp.log	NTP installation log file
	start_ntp.log	NTP startup log file
	ntpChecker.log	NTP check log file
	ntpMonitor.log	NTP monitoring log file
	heartbeat_trace.log	Log file that records heartbeats between NodeAgent and Controller
	alarm.log	Alarm log
	monitor.log	Monitoring log file
	nodeagent_ctl.log, start-agent.log	NodeAgent startup log file
	agent.log	NodeAgent run log file
	cert.log	Certificate log file
	agentplugin.log	Agent plug-in running status monitoring log file
	omaplugin.log	OMA plug-in run log file
	diskhealth.log	Disk health check log file
	supressionAlarm.log	Alarm script run log file
	updateHostFile.log	Host list update log file
	collectLog.log	Run log of the node log collection script
	host_metric_collect.log	Host index collection run log

Log Type	Log File Name	Description
	checkfileconfig.log	Run log file of file permission check
	entropycheck.log	Entropy check run log file
	timer.log	Log of scheduled node scheduling
	pluginmonitor.log	Component monitoring plug-in log
	agent_alarm_py.log	Log file that records alarms upon insufficient NodeAgent file permission
okerberos	addRealm.log, modifyKerberosRealm.log	Realm handover log file
	checkservice_detail.log	Okerberos health check log file
	genKeytab.log	keytab generation log file
	KerberosAdmin_genConfigDetail.log	Run log file of generating kadmin.conf when starting the kadmin process
	KerberosServer_genConfigDetail.log	Run log file of generating krb5kdc.conf when starting the krb5kdc process
	oms-kadmind.log	Run log file of the kadmin process
	oms_kerberos_install.log, postinstall_detail.log	Okerberos installation log file
	oms-krb5kdc.log	Run log file of the krbkdc process
	start_detail.log	Okerberos startup log file
	realmDataConfigPro-cess.log	Log file rollback for realm handover failure.
	stop_detail.log	Okerberos stop log file
oldapserver	ldapserver_backup.log	Oldapserver backup log file

Log Type	Log File Name	Description
	ldapserver_chk_service.log	Oldapserver health check log file
	ldapserver_install.log	Oldapserver installation log file
	ldapserver_start.log	Oldapserver startup log file
	ldapserver_status.log	Log file that records the status of the Oldapserver process
	ldapserver_stop.log	Oldapserver stop log file
	ldapserver_wrap.log	Oldapserver service management log file
	ldapserver_uninstall.log	Oldapserver uninstallation log file
	restart_service.log	Oldapserver restart log file
	ldapserver_unlockUser.log	Log file that records information about unlocking LDAP users and managing accounts
metric_agent	gc.log	MetricAgent JAVA VM gc log file
	metric_agent.log	Run log file of MetricAgent.
	metric_agent_qps.log	Log file that records MetricAgent Internal queue length and qps information
	metric_agent_root.log	All run logs of MetricAgent
	start.log	Log file that records information about the MetricAgent startup and stop
omm	omsconfig.log	OMS configuration log file
	check_oms_heartbeat.log	OMS heartbeat log file
	monitor.log	OMS monitoring log file

Log Type	Log File Name	Description
	ha_monitor.log	HA_Monitor operation log file
	ha.log	HA operation log file
	fms.log	Alarm log file
	fms_ha.log	HA alarm monitoring log file
	fms_script.log	Alarm control log file
	config.log	Alarm configuration log file
	iam.log	IAM log file
	iam_script.log	IAM control log file
	iam_ha.log	IAM HA monitoring log file
	config.log	IAM configuration log file
	operatelog.log	IAM operation log file
	heartbeatcheck_ha.log	OMS heartbeat HA monitoring log file
	install_oms.log	OMS installation log file
	pms_ha.log	HA monitoring log file
	pms_script.log	Monitoring control log file
	config.log	Monitoring configuration log file
	plugin.log	Monitoring plug-in run log file
	pms.log	Monitoring log file
	ha.log	HA run log file
	cep_ha.log	CEP HA monitoring log file
	cep_script.log	CEP control log file
	cep.log	CEP log file
	config.log	CEP configuration log file

Log Type	Log File Name	Description
	omm_gaussdba.log	GaussDB HA monitoring log file
	gaussdb-<SERIAL>.log	GaussDB run log file
	gs_ctl-<DATE>.log	GaussDB control log archive log file
	gs_ctl-current.log	GaussDB control log file
	gs_guc-current.log	GaussDB operation log file
	encrypt.log	Omm encryption log file
	omm_agent_ctl.log	OMA control log file
	oma_monitor.log	OMA monitoring log file
	install_oma.log	OMA installation log file
	config_oma.log	OMA configuration log file
	omm_agent.log	OMA run log file
	acs.log	ACS resource log file
	aos.log	AOS resource log file
	controller.log	Controller resource log file
	feed_watchdog.log	feed_watchdog resource log file
	floatip.log	Floating IP address resource log file
	ha_ntp.log	NTP resource log file
	httpd.log	Httpd resource log file
	okerberos.log	Okerberos resource log file
	oldap.log	OLdap resource log file
	tomcat.log	Tomcat resource log file
	send_alarm.log	Run log file of the HA alarm sending script of the management node
timestamp	restart_stamp	NodeAgent start time log file

Log Type	Log File Name	Description
tomcat	cas.log, localhost_access_cas_log.log	CAS run log file
	catalina.log, catalina.out, host-manager.log, localhost.log, manager.log	Tomcat run log file
	localhost_access_web_log.log	Log file that records the access to REST APIs of FusionInsight Manager
	web.log	Run log file of the Web process
	northbound_ftp_sftp.log, snmp.log	Northbound logs
	perfStats.log	Performance statistics log file
watchdog	watchdog.log, feed_watchdog.log	watchdog run log file
update-service	omm_upd_server.log	UPDServer run log file
	omm_upd_agent.log	UPDAgent run log file
	update-manager.log	UPDManager run log file
	install.log	Installation logs during service upgrade
	uninstall.log	Uninstallation logs during service upgrade

Log Type	Log File Name	Description
	catalina.<*Time*>.log, catalina.out, host-manager.<*Time*>.log, localhost.<*Time*>.log, manager.<*Time*>.log, manager_access_log.<*Time*>.txt, web_service_access_log.<*Time*>.txt, catalina.log, gc-update-service.log.0.current, update-manager.controller, update-web-service.controller, update-web-service.log, commit_rm_distributed.log, commit_rm_upload_package.log, common_omagent_operator.log, forbid_monitor.log, initialize_package_atoms.log, initialize_unzip_pack.log, omm-upd.log, register_patch_pack.log, resume_monitor.logrollback_clear_patch.log, unregister_patch_pack.log, update-rcommupd.log, update-rcupdatemanager.log, update-service.log	Run logs during service upgrade
upgrade	upgrade.log_<*Time*>	OMS upgrade log file
	rollback.log_<*Time*>	OMS rollback log file
sudo	sudo.log	Sudo script execution log file

Log Level

Table 10-65 describes the log levels provided by Manager. The log levels are FATAL, ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the set level are printed by programmer. The number of printed logs decreases as the set log level increases.

Table 10-65 Log levels

Severity	Description
FATAL	Logs of this level record fatal error information about the current event processing that may result in a system crash.
ERROR	Logs of this level record error information about the current event processing, which indicates that system running is abnormal.
WARN	Logs of this level record abnormal information about the current event processing. These abnormalities will not result in system faults.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record system information and system commissioning information.

Log Format

The following table lists the Manager log formats.

Table 10-66 Log format

Log Type	Component	Format	Example
Controller, Httpd, logman, NodeAgent, okerberos, oldapserver, omm, tomcat, upgrade	Controller, Httpd, logman, NodeAgent, okerberos, oldapserver, omm, tomcat, upgrade	<yyyy-MM-dd HH:mm:ss, SSS>|<Log Level>|<*Name of the thread for which the log is generated*>|<*Log message*>|<*Location where the log event occurs*>	2020-06-30 00:37:09,067 INFO [pool-1-thread-1] Completed Discovering Node. com.xxx.hadoop.om.controller.tasks.nodesetup.DiscoverNodeTask.execute(DiscoverNodeTask.java:299)

10.10.3 Configuring the Log Level and Log File Size

Scenarios

If you need to change the log level of logs, you can change the log level of FusionInsight Manager. For a specific service, you can change the log level and the log file size to prevent the failure in saving logs due to insufficient disk space.

Impact on the System

The services need to be restarted for the new configuration to take effect. During the restart, the services are unavailable.

Changing the FusionInsight Manager Log Level

1. Log in to the active management node as user **omm**.
2. Run the following command to switch to the required directory:

   cd ${BIGDATA_HOME}/om-server/om/sbin
3. Run the following command to change the log level:

   ./setLogLevel.sh*Log level parameters*

   The log level parameters are as follows and are listed in descending order by priority: FATAL, ERROR, WARN, INFO, and DEBUG. A program prints logs higher than or equal to a specified level. The higher the log level is, the fewer logs are printed.

   - **DEFAULT**: After this parameter is set, the default log level is used.
   - **FATAL**: severity of a critical error log. After this parameter is set, only logs of the FATAL level is recorded.
   - **ERROR**: error log level. After this parameter is set, logs of the ERROR and FATAL levels are displayed.
   - **WARN**: warning log level. After this parameter is set, logs of the WARN, ERROR, and FATAL levels are recorded.
   - **INFO** (default): informational log level. After this parameter is set, logs of the INFO, WARN, ERROR, and FATAL levels are displayed.
   - **DEBUG**: debugging log level. After this parameter is set, logs of the DEBUG, INFO, WARN, ERROR, and FATAL levels are displayed.
   - **TRACE**: tracing log level. After this parameter is set, logs of the TRACE, DEBUG, INFO, WARN, ERROR, and FATAL levels are displayed.

   NOTE

   The log levels of components are different as defined in open-source code.
4. Download and view logs to verify that the log level settings take effect. For details, see **Log**.

Changing the Service Log Level and Log File Size

NOTE

The KrbServer, LdapServer, and DBService do not support the modification of the service log level and log file size.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* **> Services**.

Step 3 Click a service in the service list. On the displayed page, switch to the **Configurations** tab page.

Step 4 On the displayed page, click the **All Configurations** tab. Expand the role instance displayed on the left of the page. Click **Log** of the role to be modified.

Step 5 Search for the parameter description and select the log level to be changed, or set the log file size in configuration page. The unit of the log file size is MB.

NOTICE

- The system automatically deletes logs based on the configured log size. To save more information, set the log file size a larger value. To ensure the integrity of log files, you are advised to manually back up the log files to another directory based on the actual service traffic before the log files are cleared according to clearance rules.
- Some services do not support the function of changing the log level on the GUI.

Step 6 Click **Save**. In the **Save Configuration** dialog box, click **OK**.

Step 7 Download and view logs to verify that the log level settings take effect.

----End

10.10.4 Configuring the Number of Local Backup Audit Log Files

Scenarios

Audit logs of cluster components are classified by name and stored in the **/var/log/Bigdata/audit** directory on each cluster node. The OMS automatically backs up the audit log directories at 03:00 every day.

The audit log directory on each node is compressed and named in the <*IP address of the node*>**.tar.gz** format. All compressed files are compressed and named in the <*yyyy-MM-dd_HH-mm-ss*>**.tar.gz** format and saved in the **/var/log/Bigdata/audit/bk/** directory on the active management node. In addition, the standby management node saves a copy of it.

By default, the maximum number of files that can be backed up by the OMS is 90. System administrators can configure the maximum number.

Procedure

Step 1 Log in to the active management node as user **omm**.

NOTE

Perform this operation only on the active management node. This operation is not supported on the standby management nodes; otherwise, the cluster cannot work properly.

Step 2 Run the following command to switch to the required directory:

cd ${BIGDATA_HOME}/om-server/om/sbin

Step 3 Run the following command to change the maximum number of backup audit log files to be reserved:

./modifyLogConfig.sh -m ***Maximum number of backup files that can be reserved by OMS***

The default value is 90. The value ranges from 0 to 365. The greater the value is, the larger disk space is occupied.

If the following information is displayed, the operation is successful:

```
Modify log config successfully
```

----End

10.10.5 Viewing Role Instance Logs

Scenario

FusionInsight Manager allows you to view the logs of each role instance online,

Procedure

Step 1 Log in to FusionInsight Manager.

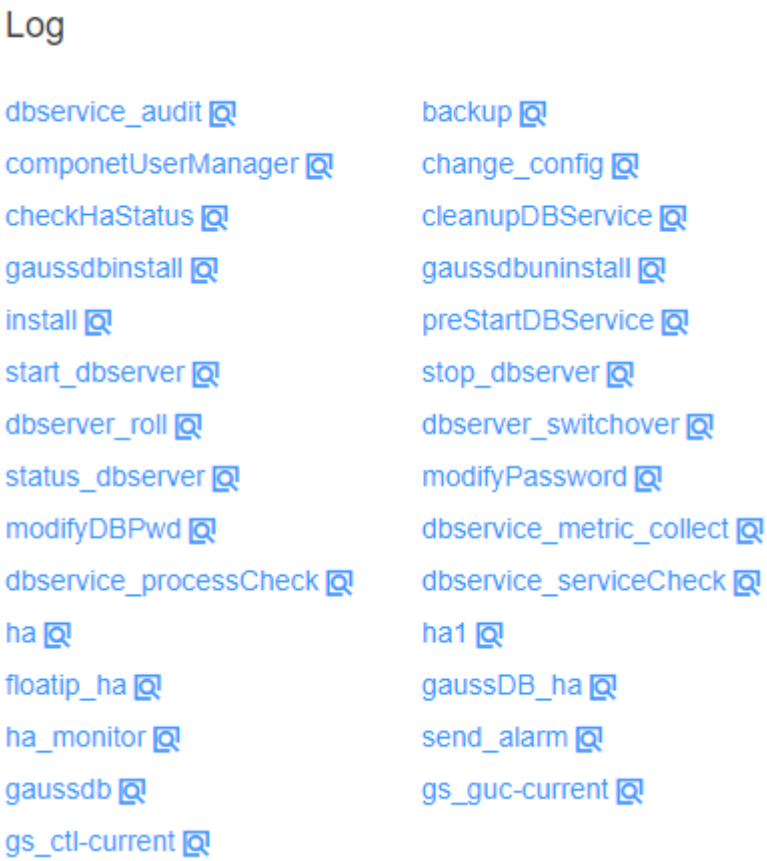
Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > *Service name* > **Instance** and click the name of the instance whose logs you want to view. The instance status page is displayed.

Step 3 In the **Log** area, click the name of the log file to be viewed to preview the log content online.

NOTE

- On the **Hosts** page, click a host name. In the **Instance** area on the host details page, you can view the log files of all role instances on the host.
- By default, a maximum of 100 lines of a log can be displayed. You can click **Load More** to view more logs. Click **Download** to download the log file to the local PC. For details about how to download service logs in batches, see **Log Downloadind**.

Figure 10-20 Viewing instance logs



----End

10.11 Backup and Recovery Management

10.11.1 Introduction

Overview

FusionInsight Manager provides backup and restoration capabilities for user data and system data in a cluster. The backup function is provided by component. The system supports backup of Manager data, component metadata (DBService, HDFS NameNode, HBase, Kafka and Yarn), and service data (HBase, HDFS, Hive).

The backup function supports data backup to the local disk, local HDFS, remote HDFS, NAS (NFS/CIFS), SFTP server and OBS. For details, see section **Backing Up Data**.

For components supporting multi-service, multiple instances of the same service can be backed up and restored. The backup and restoration operations are the same as those when there is one instance.

NOTE

MRS 3.1.0 and later versions support backing up data to OBS.

The backup and recovery tasks are performed in the following scenarios:

- Routine backup is performed to ensure the data security of the system and components.
- When the system is faulty, the data backup can be used to recover the system.
- When the active cluster is completely faulty, an image cluster same as the active cluster needs to be created, and backed up data can be used to perform restoration operations.

Table 10-67 Backing up Manager configuration data based on service requirements

Backup Type	Backup Content	Backup Directory Type
OMS	Back up database data (excluding alarm data) and configuration data in the cluster management system by default.	• LocalDir • LocalHDFS • RemoteHDFS • NFS • CIFS • SFTP • OBS

Table 10-68 Backing up component metadata or other data based on service requirements

Backup Type	Backup Content	Backup Directory Type
DBService	Back up metadata of components (including Loader, Hive, Spark, Oozie, Hue) managed by DBService. After the multi-instance function is enabled, the metadata of multiple Hive and Spark service instances is backed up.	• LocalDir • LocalHDFS • RemoteHDFS • NFS • CIFS • SFTP • OBS
Kafka	Kafka metadata.	• LocalDir • LocalHDFS • RemoteHDFS • NFS • CIFS • OBS

Backup Type	Backup Content	Backup Directory Type
NameNode	Back up HDFS metadata. For clusters enabled with multi-service, the backup and recovery function is supported for these NameServices and the backup and recovery operations are consistent with those of the default instance **hacluster**.	• LocalDir • RemoteHDFS • NFS • CIFS • SFTP • OBS
Yarn	Back up information about the Yarn service resource pool.	
HBase	**tableinfo** file and data files of HBase	

Table 10-69 Backing up service data of specific components based on service requirements

Backup Type	Backup Content	Backup Directory Type
HBase	Back up table-level user data. For clusters enabled with multi-service, the multiple HBase service instances can be backed up and restored. The backup and restoration operations are the same as those for the HBase service instance.	• RemoteHDFS • NFS • CIFS • SFTP
HDFS	Back up the directories or files that correspond to user services. **NOTE** Encrypted directories cannot be backed up or restored.	
Hive	Back up table-level user data. For clusters enabled with multi-service, the multiple Hive service instances can be backed up and restored. The backup and restoration operations are the same as those for the Hive service instance.	

Note that some components do not provide the data backup and restoration functions:

- Kafka supports copies and allows multiple copies to be specified when a topic is created.
- Mapreduce and Yarn data is stored in the HDFS. Therefore, MapReduce and Yarn depend on the HDFS to provide the backup and restoration functions.
- Backup and restoration of service data in ZooKeeper are performed by their own upper-layer components.

Principles

Task

Before backup or recovery, you need to create a backup or recovery task and set task parameters, such as the task name, backup data source, and type of backup file save path. Data backup and recovery can be performed by executing backup and recovery tasks. When the Manager is used to recover the data of HDFS, HBase, Hive, and NameNode, the cluster cannot be accessed.

Each backup task can back up data of different data sources and generates an independent backup file for each data source. All the backup files generated in each backup task form a backup file set, which can be used in recovery tasks. Backup data can be stored on Linux local disks, local cluster HDFS, and standby cluster HDFS.

The backup task provides the full backup or incremental backup policies. Metadata data backup tasks do not support incremental backup policie. If the backup path type is NFS or CIFS, incremental backup is not recommended. When an incremental backup is used for NFS or CIFS backup, the backup data of the latest full backup is updated for each incremental backup. Therefore, no new recovery point is generated.

NOTE

Task execution rules:

- If a task is being executed, the task cannot be executed repeatedly and other tasks cannot be started too.
- The interval at which a periodical task is automatically executed must be greater than 120s; otherwise, the task is postponed and executed in the next period. Manual tasks can be executed at any interval.
- When a period task is to be automatically executed, the current time cannot be 120s later than the task start time; otherwise, the task is postponed and executed in the next period.
- When a periodical task is locked, it cannot be automatically executed and needs to be manually unlocked.
- Before an OMS, LdapServer, Kafka or NameNode backup task starts, ensure that the **LocalBackup** partition on the active management node has more than 20 GB available space; otherwise, the backup task cannot be started.

When planning backup and recovery tasks, select the data to be backed up or recovered strictly based on the service logic, data store structure, and database or table association. By default, the system creates the periodic backup tasks **default-oms** and **default**-*cluster ID* at an interval of one hour, to fully back up OMS and metadata of DBService and NameNode to the local disk.

Snapshot

The system adopts snapshot technology to quickly back up data. Snapshots include HBase snapshots HDFS snapshots.

- HBase snapshot

  An HBase snapshot is a backup file of HBase tables at a specified time point. This backup file does not copy service data or affect the RegionServer. The HBase snapshot copies table metadata, including table descriptor, region info, and HFile reference information. The metadata can be used to recover data before the snapshot creation time.

- HDFS snapshot

  An HDFS snapshot is a read-only backup copy of the HDFS file system at a specified time point. The snapshot is used in data backup, misoperation protection, and disaster recovery scenarios.

  The snapshot function can be enabled for any HDFS directory to create the related snapshot file. Before creating a snapshot for a directory, the system automatically enables the snapshot function for the directory. Creating a snapshot does not affect any HDFS operation. A maximum of 65536 snapshots can be created for each HDFS directory.

  When a snapshot is being created for an HDFS directory, the directory cannot be deleted or modified before the snapshot is created. Snapshots cannot be created for the upper-layer directories or subdirectories of the directory.

DistCp

Distributed copy (DistCp) is a tool used to perform large-amount data replication in the cluster HDFS or between the HDFSs of different clusters. In an HBase, HDFS or Hive metadata backup or recovery task, if the data is backed up in the HDFS of the standby cluster, the system invokes DistCp to perform the operation. Install the MRS system of the same version on the active and standby clusters.

DistCp uses Mapreduce to implement data distribution, troubleshooting, recovery, and report. DistCp specifies different Map jobs for various source files and directories in the specified list. Each Map job copies the data in the partition that corresponds to the specified file in the list.

To use DistCp to perform data replication between the HDFS of two clusters, configure the trust relationship and cross-cluster replication function for both clusters (The mutual trust relationship does not need to be configured for clusters managed by the same FusionInsight Manager.). When backing up the cluster data to HDFS in another cluster, you need to install the Yarn component. Otherwise, the backup fails.

Local rapid recovery

After using DistCp to back up the HBase, HDFS, and Hive data of the local cluster in the HDFS of the standby cluster, the HDFS of the local cluster retains the backup data snapshots. Users can create local rapid recovery tasks to recovery day by using the snapshot files in the HDFS of the local cluster.

NAS

Network Attached Storage (NAS) is a dedicated data storage server which includes the storage device and embedded system software. It provides the cross-platform file sharing function. By using NFS (supporting NFSv3 and NFSv4) and CIFS (supporting SMBv2 and SMBv3) protocols, users can connect the FusionInsight service plane with the NAS server to back up or restore data to or from the NAS.

NOTE

- Before data is backed up to the NAS, the system automatically mounts the NAS shared address to a local partition. After the backup is complete, the system uninstalls the NAS shared partition.
- To prevent backup and restoration failures, do not access the shared address where the NAS server mounts to the local host during data backup and restoration, for example, **/srv/BigData/LocalBackup/nas**.
- When service data is backed up to the NAS, DistCp is used.

Specifications

Table 10-70 Backup and recovery feature specifications

Item	Parameter
Maximum number of backup or recovery tasks in a cluster	100
Number of concurrent running tasks	1
Maximum number of waiting tasks	199
Maximum size of backup files on a Linux local disk (GB)	600

NOTE

If service data is stored in the ZooKeeper upper-layer components, ensure that the number of znodes in a single backup or restoration task is not too large. Otherwise, the task will fail, and the ZooKeeper service performance will be affected. To check the number of znodes in a single backup or restoration task, perform as follows:

- Ensure that the number of znodes in a single backup or restoration task is less than the upper limit of OS file handles.
  1. To check the upper limit at the system level, run the **cat /proc/sys/fs/file-max** command.
  2. To check the upper limit at the user level, run the **ulimit -n** command.
- If the number of znodes in the parent directory exceeds the upper limit, back up and restore data in its sub-directories in batches. To check the number of znodes using ZooKeeper client scripts, perform as follows:
  1. On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Zookeeper** > **Instance** and view the management IP address of each ZooKeeper role.
  2. Log in to the node where the client resides and run the following command:

     zkCli.sh -server *ip*:*port*, where, the IP address can be any management IP address, and the default port number is 2181.
  3. If the following information is displayed, login to the ZooKeeper server succeeds:
     ```
     WatchedEvent state:SyncConnected type:None path:null
     [zk: ip:port(CONNECIED) 0]
     ```
  4. Run the **getusage** command to check the number of znodes in the directory to be backed up. For example:

     getusage /hbase/region. In the command output, **Node count** indicates the number of znodes stored in the **region** directory.

Table 10-71 Specifications of the **default** task

Item	OMS	HBase	Kafka	DBService	NameNode
Backup period	1 hour				
Maximum number of copies	168 (Historical records of seven days)				24 (Historical records of one day)
Maximum size of a backup file	10 MB	10 MB	512MB	100 MB	20 GB
Maximum size of disk space used	1.64 GB	1.64 GB	84GB	16.41 GB	480 GB
Save path of backup data	*Data path*/**LocalBackup/** on active and standby management nodes				

NOTE

- The administrator must regularly transfer the backup data of the default task to an external cluster based on the enterprise's O&M requirements.
- The administrator can create a DistCp backup task to store data of OMS, DBService, and NameNode to an external cluster.
- The running duration of a cluster data backup task can be calculated based on the volume of data to be backed up divided by the network bandwidth between the cluster and backup device. In actual scenarios, you are advised to multiply the calculated duration by 1.5 as a reference value.
- Performing a data backup task affects the maximum I/O performance of the cluster. Therefore, it is recommended that the backup task run time be staggered from the cluster peak hours.

10.11.2 Backing Up Data

10.11.2.1 Backing Up OMS Data

Scenario

To ensure FusionInsight Manager data security routinely or before and after a critical operation (such as capacity expansion and reduction) on Manager, Manager data needs to be backed up. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up Manager data. Both automatic backup tasks and manual backup tasks are supported.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan the backup type, period, policy, and other specifications, and check whether ***Data path***/LocalBackup/ has sufficient space on active and standby management nodes.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.

- If you want to back up data to the OBS, ensure that the current cluster is connected to OBS and you have the permission to access OBS.

Procedure

Step 1 On FusionInsight Manager, choose **O&M > Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Set **Backup Object** to **OMS**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-72 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 Set **Configuration** to **OMS**.

Step 7 Set **Path Type** of **OMS** to a backup directory type.

The following backup directory types are supported:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node and the standby management node automatically synchronizes the backup file.

  The default save path is *Cluster data path*/**LocalBackup/**, for example, **/srv/BigData/LocalBackup**.

If you select this value, you need to set **Maximum Number of Backup Copies** to specify the number of backup files that can be retained in the backup directory.

- **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster.

  If you select this value, you need to set the following parameters:

  - **Target Path**: indicates the backup file save path in the HDFS. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Cluster for Backup**: Enter the cluster name mapping to the backup directory.
  - **Target NameService Name**: indicates the NameService name that corresponds to the backup directory. The default value is **hacluster**.

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Source Cluster**: Select the cluster of the Yarn queue used by the backup data from **Source Cluster**.
  - **Queue Name**: indicates the name of the Yarn queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the source cluster.

- **NFS:** indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.

  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **CIFS:** indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **OBS**: indicates that the backup files are stored in the OBS.

  If you select this value, you need to set the following parameters:
  - **Target Path**: indicates the OBS directory for storing backup data.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

    NOTE

    MRS 3.1.0 and later versions support backing up data to OBS.

Step 8 Click **OK** to save the settings.

Step 9 In the **Operation** column of the created task in the backup task list, choose **More > Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory.

The format of the subdirectory name is *backup task name_task creation time*, and the subdirectory is used to save data source backup files.

The format of the backup file name is *version_data source_task execution time*.**tar.gz**.

----End

10.11.2.2 Backing Up DBService Data

Scenario

To ensure system data security routinely or before and after a critical operation (such as upgrade and migration) on DBService, DBService data needs to be backed up. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up DBService data. Both automatic backup tasks and manual backup tasks are supported.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan the backup type, period, policy, and other specifications, and check whether ***Data path***/LocalBackup/ has sufficient space on active and standby management nodes.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.
- If you want to back up data to the OBS, ensure that the current cluster is connected to OBS and you have the permission to access OBS.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-73 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 Set **Configuration** to **DBService**.

NOTE

If there are multiple DBServices, all DBServices are backed up by default. You can click **Assign Service** to specify the DBService to be backed up.

Step 7 Set **Path Type** of **DBService** to a backup directory type.

The following backup directory types are supported:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node and the standby management node automatically synchronizes the backup file.

  The default save path is *Cluster data path*/**LocalBackup/**, for example, **/srv/BigData/LocalBackup**.

  If you select this value, you need to set **Maximum Number of Backup Copies** to specify the number of backup files that can be retained in the backup directory.
- **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster.

If you select this value, you need to set the following parameters:

- **Target Path**: indicates the backup file save path in the HDFS. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
- **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **Target NameService name**: indicates the NameService name that corresponds to the backup directory. The default value is **hacluster**.

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the source cluster.

- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.

  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **OBS**: indicates that the backup files are stored in the OBS.

  If you select this value, you need to set the following parameters:
  - **Target Path**: indicates the OBS directory for storing backup data.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

    NOTE

    MRS 3.1.0 and later versions support backing up data to OBS.

Step 8 Click **OK** to save the settings.

Step 9 In the **Operation** column of the created task in the backup task list, choose **More > Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_task creation time*, and the subdirectory is used to save data source backup files. The format of the backup file name is *version_data source_task execution time*.tar.gz.

----End

10.11.2.3 Backing Up HBase Metadata

Scenario

To avoid that the HBase service becomes unavailable when the HBase system table directory and files are corrupted or after a system administrator performs a critical operation (such as upgrade and migration) on HBase, HBase metadata (tableinfo and HFile) needs to be backed up to ensure security. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up HBase metadata. Both automatic backup tasks and manual backup tasks are supported.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan backup policies, such as the backup task type, period, backup object, and YARN queue that required by the backup task.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.
- The **fs.defaultFS** parameter of HBase must be the same as that of Yarn and HDFS.
- If HBase data is stored in the local HDFS, HBase metadata can be backed up to OBS. If HBase data is stored on OBS, data backup is not supported.
- If you want to back up data to the OBS, ensure that the current cluster is connected to OBS and you have the permission to access OBS.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-74 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 In **Configuration**, select **HBase** under **Metadata and other data**.

NOTE

If there are multiple HBase services, all HBase services are backed up by default. You can click **Assign Service** to specify the HBase to be backed up.

Step 7 Set **Path Type** of **HBase** to a backup directory type.

The following backup directory types are supported:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node and the standby management node automatically synchronizes the backup file.

  The default save path is *Cluster data path*/**LocalBackup/**, for example, **/srv/BigData/LocalBackup**.

  If you select this value, you need to set **Maximum Number of Backup Copies** to specify the number of backup files that can be retained in the backup directory.
- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster. If you select this value, you need to set the following parameters:
  - **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in

remote cluster of the cluster, or the NameService name of a configured remote cluster.

- **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
- **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
- **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
- **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the source cluster.

- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **OBS**: indicates that the backup files are stored in the OBS.

  If you select this value, you need to set the following parameters:
  - **Target Path**: indicates the OBS directory for storing backup data.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

    NOTE

    MRS 3.1.0 and later versions support backing up data to OBS.

Step 8 Click **OK** to save the settings.

Step 9 In the **Operation** column of the created task in the backup task list, choose **More > Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_task creation time*, and the subdirectory is used to save data source backup files. The format of the backup file name is *version_data source_task execution time*.tar.gz.

----End

10.11.2.4 Backing Up HBase Service Data

Scenario

To ensure system data security routinely or before and after a critical operation (such as upgrade and migration) on HBase, HBase data needs to be backed up. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up HBase data. Both automatic backup tasks and manual backup tasks are supported.

The following scenarios may occur when HBase backs up data:

- When a user creates an HBase table, **KEEP_DELETED_CELLS** is set to **false** by default. When the user backs up this HBase table, deleted data will be backed

up and junk data may exist after data restoration. Based on service requirements, this parameter needs to be set to **true** manually when an HBase table is created.

- When a user manually specifies the time stamp when writing data into an HBase table and the specified time is earlier than the last backup time of the HBase table, new data may not be backed up in incremental backup tasks.
- The HBase backup function does not support backing up the access control lists (ACLs) of read, write, create, execute, and administrate operations on HBase globals or namespaces. After the HBase data is restored, the administrator needs to set new permission for roles on FusionInsight Manager.
- Assume that an HBase backup task has been created and the current backup data in the standby cluster is lost. The next incremental task will fail and a new HBase backup task needs to be created. The next full backup task will be normal.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan backup policies, such as the backup task type, period, backup object, and YARN queue that required by the backup task.
- Check whether HDFS of the standby cluster has sufficient space. It is recommended the directory for storing backup files be a user-defined directory.
- On the HDFS client, run **hdfs lsSnapshottableDir** as user **hdfs** to check the list of directories for which HDFS snapshots have been created in the current cluster. Ensure that the HDFS parent directory or subdirectory where data files to be backed up are stored does not have HDFS snapshots. Otherwise, the backup task cannot be created.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.
- The **fs.defaultFS** parameter of HBase must be the same as that of Yarn and HDFS.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration> Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-75 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 In **Configuration**, select **HBase > HBase** under **Service Data**.

Step 7 Set **Path Type** of **HBase** to a backup directory type.

The following backup directory types are supported:

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  – **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.

  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **10**

Step 8 Set **Maximum Number of Recovery Points** to the number of snapshots that can be retained in the cluster.

Step 9 Set **Backup Content** to one or multiple HBase tables to be backed up.

The following methods are supported to select backup data:

- Select directly

  Click the name of a naming space in the navigation tree to show all the tables in the naming space, and select specified tables.
- Select using regular expressions
  - Click **Query Regular Expression**.

- Enter the naming space where the HBase tables are located in the first text box as prompted. The naming space must be the same as the existing naming space, for example, **default**.
- Enter a regular expression in the second text box. Standard regular expressions are supported. For example, if all tables in the database need to be filtered, enter **([\s\S]*?)**. If tables of which the names consisting of letters and digits, such as **tb***1*, need to be filtered, enter **tb\d***.
- Click **Refresh** to view the selected tables in **Directory Name**.
- Click **Synchronize** to save the result.

NOTE

- When entering regular expressions, you can click + or − to add or delete an expression.
- If the selected table or directory is incorrect, click **Clear Selected Node** to deselect it.

Step 10 Click **Verify** to check whether the backup task is configured correctly.

The possible causes of check failure are as follows:

- Target NameNode IP address is incorrect.
- The queue name is incorrect.
- The HDFS parent directory or subdirectory where HBase table data files to be backed up are stored has HDFS snapshots.
- The directory or table to be backed up does not exist.

Step 11 Click **OK** to save the settings.

Step 12 In the **Operation** column of the created task in the backup task list, choose **More** > **Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_data source_task creation time*, and the subdirectory is used to save latest data source backup files. All the backup file sets are saved to the related snapshot directories.

----End

10.11.2.5 Backing Up NameNode Data

Scenario

To ensure system data security routinely or before and after a critical operation (such as upgrade and migration) on NameNode, NameNode data needs to be backed up. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up NameNode data. Both automatic backup tasks and manual backup tasks are supported.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan the backup type, period, policy, and other specifications, and check whether ***Data path***/LocalBackup/ has sufficient space on active and standby management nodes.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.
- If you want to back up data to the OBS, ensure that the current cluster is connected to OBS and you have the permission to access OBS.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-76 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.

Parameter Name	Description
Backup Policy	Only **Full backup every time** is supported. **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 Set **Configuration** to **NameNode**.

Step 7 Set **Path Type** of **NameNode** to a backup directory type.

The following backup directory types are supported:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node and the standby management node automatically synchronizes the backup file. The default save path is ***Data path*/LocalBackup/**.

  If you select this value, you need to set the following parameters:

  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
- **NFS:** indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.

  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **OBS**: indicates that the backup files are stored in the OBS.

  If you select this value, you need to set the following parameters:
  - **Target Path**: indicates the OBS directory for storing backup data.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.

    NOTE

    MRS 3.1.0 and later versions support backing up data to OBS.

Step 8 Click **OK** to save the settings.

Step 9 In the **Operation** column of the created task in the backup task list, choose **More** > **Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_task creation time*, and the subdirectory is used to save data source backup files. The format of the backup file name is *version_data source_task execution time*.tar.gz.

----End

10.11.2.6 Backing Up HDFS Service Data

Scenario

To ensure system data security routinely or before and after a critical operation (such as upgrade and migration) on HDFS, HDFS data needs to be backed up. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up HDFS data. Both automatic backup tasks and manual backup tasks are supported.

NOTE

Encrypted directories cannot be backed up or restored.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan backup policies, such as the backup task type, period, backup object, and YARN queue that required by the backup task.
- Check whether HDFS of the standby cluster has sufficient space. It is recommended the directory for storing backup files be a user-defined directory.
- On the HDFS client, run **hdfs lsSnapshottableDir** as user **hdfs** to check the list of directories for which HDFS snapshots have been created in the current cluster. Ensure that the HDFS parent directory or subdirectory where data files to be backed up are stored does not have HDFS snapshots. Otherwise, the backup task cannot be created.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-77 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.

Parameter Name	Description
Period	Indicates task execution interval. The options include **Hours** and **Days**.
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 Set **Configuration** to **HDFS**.

Step 7 Set **Path Type** of **HDFS** to a backup directory type.

The following backup directory types are supported:

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.

  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.

  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**

- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**

Step 8 Set **Maximum Number of Recovery Points** to the number of snapshots that can be retained in the cluster.

Step 9 Set **Backup Content** to one or multiple HDFS directories to be backed up based on service requirements.

The following methods are supported to select backup data:

- Select directly

  Click the name of a directory in the navigation tree to show all the subdirectories in the directory, and select specified directories.

- Select using regular expressions
  - Click **Query Regular Expression**.
  - Enter the parent directory full path of the directory in the first text box as prompted. The directory must be the same as the existing directory, for example, **/tmp**.
  - Enter a regular expression in the second text box. Standard regular expressions are supported. For example, if all files or subdirectories in the parent directory need to be filtered, enter **([\s\S]*?)**. If files of which the names consisting of letters and digits, such as **file***1*, need to be filtered, enter **file\d***.

  - Click **Refresh** to view the selected tables in **Directory Name**.
  - Click **Synchronize** to save the result.

  NOTE

  - When entering regular expressions, you can click + or — to add or delete an expression.
  - If the selected table or directory is incorrect, click **Clear Selected Node** to deselect it.
  - The backup directory cannot contain files that have been written for a long time. Otherwise, the backup task will fail. Therefore, you are advised not to perform operations on the top-level directory, such as **/user**, **/tmp**, and **/mr-history**.

Step 10 Click **Verify** to check whether the backup task is configured correctly.

The possible causes of check failure are as follows:

- Target NameNode IP address is incorrect.
- The queue name is incorrect.
- The HDFS parent directory or subdirectory where data files to be backed up are stored has HDFS snapshots.
- The directory or table to be backed up does not exist.
- The name of the NameService is incorrect.

Step 11 Click **OK** to save the settings.

Step 12 In the **Operation** column of the created task in the backup task list, choose **More > Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_data source_task creation time*, and the subdirectory is used to save latest data source backup files. All the backup file sets are saved to the related snapshot directories.

----End

10.11.2.7 Backing Up Hive Service Data

Scenario

To ensure system data security routinely or before and after a critical operation (such as upgrade and migration) on Hive, Hive data needs to be backed up. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up Hive data. Both automatic backup tasks and manual backup tasks are supported.

- The Hive backup and recovery function cannot identify the service and structure relationships of objects such as Hive tables, indexes, and views. When executing backup and recovery tasks, the user needs to manage a unified recovery point based on the service scenario to ensure proper service running.

- The Hive backup and recovery function does not support Hive on RDB data tables. The original data tables need to be backed up and recovered in the external database independently.
- Assume that a Hive backup task has been created and includes Hive on HBase tables, and the current backup data in the standby cluster is lost. The next incremental task will fail and a new Hive backup task needs to be created. The next full backup task will be normal.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.
- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan backup policies, such as the backup task type, period, backup object, and YARN queue that required by the backup task.
- Check whether HDFS of the standby cluster has sufficient space. It is recommended the directory for storing backup files be a user-defined directory.
- On the HDFS client, run **hdfs lsSnapshottableDir** as user **hdfs** to check the list of directories for which HDFS snapshots have been created in the current cluster. Ensure that the HDFS parent directory or subdirectory where data files to be backed up are stored does not have HDFS snapshots. Otherwise, the backup task cannot be created.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-78 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 Set **Configuration** to **Hive > Hive**.

Step 7 Set **Path Type** of **Hive** to a backup directory type.

The following backup directory types are supported:

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Target Path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Server shared Path**: indicates the configured shared directory on the NAS server.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.

  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **SFTP:** Indicates that backup files are stored in the server using SFTP.

  If you select SFTP, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**:Enter the password for connecting to the server using SFTP.
  - **Server Shared Path**: Enter the backup path on the SFTP server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **NameService Name**: indicates the NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**

Step 8 Set **Maximum Number of Recovery Points** to the number of snapshots that can be retained in the cluster.

Step 9 Set **Backup Content** to one or multiple Hive tables to be backed up.

The following methods are supported to select backup data:

- Select directly

  Click the name of a database in the navigation tree to show all the tables in the database, and select specified tables.
- Select using regular expressions
  - Click **Query Regular Expression**.
  - Enter the database where the Hive tables are located in the first text box as prompted. The database must be the same as the existing database, for example, **default**.

- Enter a regular expression in the second text box. Standard regular expressions are supported. For example, if all tables in the database need to be filtered, enter **([\s\S]*?)**. If tables of which the names consisting of letters and digits, such as **tb***1*, need to be filtered, enter **tb\d***.
- Click **Refresh** to view the selected tables in **Directory Name**.
- Click **Synchronize** to save the result.

NOTE

- When entering regular expressions, you can click + or — to add or delete an expression.
- If the selected table or directory is incorrect, click **Clear Selected Node** to deselect it.

Step 10 Click **Verify** to check whether the backup task is configured correctly.

The possible causes of check failure are as follows:

- Target NameNode IP address is incorrect.
- The queue name is incorrect.
- The HDFS parent directory or subdirectory where data files to be backed up are stored has HDFS snapshots.
- The directory or table to be backed up does not exist.

Step 11 Click **OK** to save the settings.

Step 12 In the **Operation** column of the created task in the backup task list, choose **More > Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_data source_task creation time*, and the subdirectory is used to save latest data source backup files. All the backup file sets are saved to the related snapshot directories.

----End

10.11.2.8 Backing Up Kafka Metadata

Scenario

To ensure Kafka metadata security or before and after a critical operation (such as upgrade and migration) on ZooKeeper, Kafka metadata needs to be backed up. The backup data can be used to recover the system in time if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a backup task in FusionInsight Manager to back up Kafka metadata. Both automatic backup tasks and manual backup tasks are supported.

Prerequisites

- If you want to back up data to the remote HDFS. A standby cluster for backing up data has been created. The mode of the standby cluster is the same as that of the active cluster. In other backup modes, you do not need to prepare the standby cluster.

- If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Based on the service requirements, plan the backup type, period, policy, and other specifications, and check whether ***Data path***/LocalBackup/ has sufficient space on active and standby management nodes.
- If you want to back up data to the NAS, you need to deploy the NAS server in advance.
- If you want to back up data to the OBS, ensure that the current cluster is connected to OBS and you have the permission to access OBS.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 Click **Create**.

Step 3 Set **Name** to the name of the backup task.

Step 4 Select the cluster to be operated from **Backup Object**.

Step 5 Set **Mode** to the type of the backup task.

Periodic indicates that the backup task is periodically executed and **Manual** indicates that the backup task is manually executed.

Table 10-79 Periodic backup parameters

Parameter Name	Description
Started	Indicates the time when the task is started for the first time.
Period	Indicates task execution interval. The options include **Hours** and **Days**.

Parameter Name	Description
Backup Policy	• **Full backup at the first time and incremental backup subsequently** • **Full backup every time** • **Full backup once every n time** **NOTE** • Incremental backup is not supported when Manager data and component metadata are backed up. Only **Full backup every time** is supported. • If **Mode** is set to **Periodic** and the **Path Type** is set to **NFS** or **CIFS**, the incremental backup function cannot be used. If incremental backup is used in this scenario, data in full backup will be updated in each incremental backup, and no recovery point will be generated.

Step 6 Set **Configuration** to **Kafka**.

NOTE

If there are multiple Kafka services, all Kafka services are backed up by default. You can click **Assign Service** to specify the Kafka to be backed up.

Step 7 Set **Path Type** of **Kafka** to a backup directory type.

The following backup directory types are supported:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node and the standby management node automatically synchronizes the backup file. The default save path is *Data path*/**LocalBackup/**.

  If you select this value, you need to set **Maximum number of backup copies** to specify the number of backup files that can be retained in the backup directory.
- **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster.

  If you select this value, you need to set the following parameters:

  – **Target path**: indicates the backup file save path in the HDFS. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.

  – **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

  – **Target NameService Name**: indicates the NameService name that corresponds to the backup directory. The default value is **hacluster**.
- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  – **Destination NameService Name**: indicates the NameService name of the standby cluster. You can set it to the NameService name (**haclusterX**,

haclusterX1, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.

- **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
- **Target NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
- **Target path**: indicates the HDFS directory for storing standby cluster backup data. The save path cannot be an HDFS hidden directory, such as snapshot or recycle bin directory, or a default system directory, such as **/hbase** or **/user/hbase/backup**.
- **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.

- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select CIFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP Address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Server Shared Path**: indicates the configured shared directory on the NAS server.
  - **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
- **OBS**: indicates that the backup files are stored in the OBS.

  If you select this value, you need to set the following parameters:

- **Target Path**: indicates the OBS directory for storing backup data.
- **Maximum Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.

  NOTE

  MRS 3.1.0 and later versions support backing up data to OBS.

Step 8 Click **OK** to save the settings.

Step 9 In the **Operation** column of the created task in the backup task list, choose **More > Back Up Now** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *backup task name_task creation time*, and the subdirectory is used to save data source backup files. The format of the backup file name is *version_data source_task execution time*.tar.gz.

----End

10.11.3 Recovering Data

10.11.3.1 Recovering OMS Data

Scenario

Manager data needs to be recovered in the following scenarios: data is modified or deleted unexpectedly and needs to be restored. After an administrator performs critical data adjustment in FusionInsight Manager, an exception occurs or the operation has not achieved the expected result. All modules are faulty and become unavailable.

System administrators can create a recovery task in FusionInsight Manager to recover Manager data. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the Manager data that is generated after the data backup and before the data recovery will be lost.

Impact on the System

- In the recovery process, the Controller needs to be restarted and FusionInsight Manager cannot be logged in or operated during the restart.
- In the recovery process, all clusters need to be restarted and cannot be accessed during the restart.

- After Manager data recovery, the data, such as system configuration, user information, alarm information, and audit information, that is generated after the data backup and before the data recovery will be lost. This may result in data query failure or cluster access failure.
- After the Manager data is recovered, the system forces the LdapServer of each cluster to synchronize data from the OLadp.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- The status of the OMS resources and the LdapServer instances of each cluster is normal. If the status is abnormal, data recovery cannot be performed.
- The status of the cluster hosts and services is normal. If the status is abnormal, data recovery cannot be performed.
- The cluster host topologies during data recovery and data backup are the same. If the topologies are different, data recovery cannot be performed and you need to back up data again.
- The services added to the cluster during data recovery and data backup are the same. If the topologies are different, data recovery cannot be performed and you need to back up data again.
- The upper-layer applications that depend on the MRS cluster are stopped.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Restoration Management > Create**.

Step 4 Set **Task Name** to the name of the recovery task.

Step 5 Set **Recovery Object** to **OMS**.

Step 6 Select **OMS**.

Step 7 Set **Path Type** of **OMS** to a backup directory type.

The settings vary according to backup directory types:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node.

  If you select this value, you need to set **Source Path** to select the backup file, for example, *version_data source_task execution time*.**tar.gz**.

- **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster.

  If you select this value, you need to set the following parameters:

  - **Source Path**: indicates the full path of the backup file in HDFS, for example, ***backup path/backup task name_task creation time/version_data source_task execution time.tar.gz***.
  - **Cluster for Restoration**: Enter the cluster name mapping to the restoration directory.
  - **Source NameService Name**: indicates the NameService name that corresponds to the backup directory when the recovery task is executed. The default value is **hacluster**.

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Source Cluster**: Select the cluster of the Yarn queue used by the recovery data from **Source Cluster**.
  - **Queue Name**: indicates the name of the Yarn queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.

- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select **NFS**, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **OBS**: indicates that the backup files are stored in the OBS directory.

  If you select this value, you need to set the following parameters:
  - **Source Path**: indicates the full path for storing backup data in OBS. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.

    NOTE

    MRS 3.1.0 and later versions support saving backup files to OBS.

Step 8 Click **OK** to save the settings.

Step 9 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

Step 10 Log in to the active and standby management nodes as user **omm**.

Step 11 Run the following command to restart OMS:

sh ${BIGDATA_HOME}/om-server/om/sbin/restart-oms.sh

The command is run successfully if the following information is displayed:

```
start HA successfully.
```

Run **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** to check whether **HAAllResOK** of the management node is **Normal** and whether FusionInsight Manager can be logged in again. If yes, the restart is successful.

Step 12 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services > KrbServer** > **More** > **Synchronize Configuration**, click **OK**, and wait for the KrbServer configuration to be synchronized and the service to be restarted.

Step 13 Choose **Cluster** > *Name of the desired cluster* > **More** > **Synchronize Configuration**s, click **OK**, and wait for the cluster configuration to be synchronized.

Step 14 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **More** > **Restart**. In the displayed dialog box, enter the password of the current login user and click **OK**. Wait for the cluster to be restarted.

----End

10.11.3.2 Recovering DBService Data

Scenario

DBService data needs to be recovered in the following scenarios: data is modified or deleted unexpectedly and needs to be restored. After an administrator performs critical data adjustment in DBService, an exception occurs or the operation has not achieved the expected result. All modules are faulty and become unavailable. Data is migrated to a new cluster.

System administrators can create a recovery task in FusionInsight Manager to recover DBService data. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the DBService data that is generated after the data backup and before the data recovery will be lost.
- By default, MRS uses DBService to store the metadata of Hive, Hue, Loader, Spark, Oozie. Recovering DBService data will recover the metadata of all these components.

Impact on the System

- After the data is recovered, the data produced between the backup time and restoration time is lost.
- After the data is recovered, the configuration of the components that depend on DBService may expire and these components need to be restarted.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- The status of the active and standby DBService instances is normal. If the status is abnormal, data recovery cannot be performed.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose O&M > **> Backup and Restoration > Restoration Management**.

Step 4 Click **Create**.

Step 5 Select the cluster to be operated from **Recovery Object**.

Step 6 Set **Task Name** to the name of the recovery task.

Step 7 Select **DBService**.

NOTE

If there are multiple DBServices, you can specify the DBService to be recovered.

Step 8 Set **Path Type** of **DBService** to a backup directory type.

The settings vary according to backup directory types:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node.

  If you select this value, you need to set **Source Path** to select the backup file, for example, ***version_data source_task execution time.tar.gz***.

- **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster.

  If you select this value, you need to set the following parameters:

  - **Source Path**: indicates the full path of the backup file in HDFS, for example, ***backup path/backup task name_task creation time/version_data source_task execution time.tar.gz***.
  - **Source NameService Name**: indicates the NameService name that corresponds to the backup directory when the recovery task is executed. The default value is **hacluster**.

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster. If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.

- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select **NFS**, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **OBS**: indicates that the backup files are stored in the OBS directory.

  If you select this value, you need to set the following parameters:

  - **Source Path**: indicates the full path for storing backup data in OBS. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.

    NOTE

    MRS 3.1.0 and later versions support saving backup files to OBS.

Step 9 Click **OK** to save the settings.

Step 10 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

----End

10.11.3.3 Recovering HBase Metadata

Scenario

To avoid that the HBase service becomes unavailable when the HBase system table directory and files are corrupted or after a system administrator performs a critical operation (such as upgrade and migration) on HBase, HBase metadata (tableinfo and HFile) needs to be backed up to ensure security. The backup data can be used to recover the system if an exception occurs or the operation has not achieved the expected result, minimizing the adverse impact on services.

System administrators can create a recovery task in FusionInsight Manager to recover HBase metadata. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the HBase data that is generated after the data backup and before the data recovery will be lost.
- It is recommended that a data restoration task restore the metadata of only one component to prevent the data restoration of other components from being affected by stopping a service or instance. If data of multiple components is restored at the same time, data restoration may fail.

  HBase metadata cannot be restored at the same time as NameNode metadata. As a result, data restoration fails.

Impact on the System

- Before recovering metadata, you need to stop the HBase service, during which the HBase upper-layer applications are unavailable.
- After the metadata is recovered, the data produced between the backup time and restoration time is lost.
- After the metadata is recovered, the HBase upper-layer applications need to be started.

Prerequisites

- If the active cluster employs the security mode, cross-cluster trust relationship has been configured for the active and standby clusters. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster employs the normal mode, no cross-cluster trust relationship is required.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- Check the directory for saving the HBase backup file.
- Stop the HBase upper-layer applications.
- You have logged in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration** > **Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Restoration Management**.

Step 4 Click **Create**.

Step 5 Set **Task Name** to the name of the recovery task.

Step 6 Select the cluster to be operated from **Recovery Object**.

Step 7 In **Restoration Configuration**, select **HBase** under **Metadata and other data**.

NOTE

If there are multiple HBae services, you can specify the HBase to be recovered.

Step 8 Set **Path Type** of **HBase** to a backup directory type.

The following backup directory types are supported:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node. If you select this value, you need to set **Source Path** to select the backup file, for example, ***version_data source_task execution time.tar.gz***.
- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster. If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select **NFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.

  - **Server IP address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **OBS**: indicates that the backup files are stored in the OBS directory.

  If you select this value, you need to set the following parameters:
  - **Source Path**: indicates the full path for storing backup data in OBS. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.

    NOTE

    MRS 3.1.0 and later versions support saving backup files to OBS.

Step 9 Click **OK** to save the settings.

Step 10 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

----End

10.11.3.4 Recovering HBase Service Data

Scenario

HBase data needs to be recovered in the following scenarios: data is modified or deleted unexpectedly and needs to be restored. After an administrator performs critical data adjustment in HBase, an exception occurs or the operation has not achieved the expected result. All modules are faulty and become unavailable. Data is migrated to a new cluster.

System administrators can create a recovery task in FusionInsight Manager to recover HBase data. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the HBase data that is generated after the data backup and before the data recovery will be lost.

Impact on the System

- During the data recovery process, the system disables the HBase table to be recovered and the table cannot be accessed in this moment. The data recovery process takes several minutes, during which the HBase upper-layer applications are unavailable.
- During data recovery, user authentication stops and users cannot create new connections.
- After the data is recovered, the data produced between the backup time and restoration time is lost.
- After the data is recovered, the HBase upper-layer applications need to be started.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Check the directory for saving the HBase backup file.
- Stop the HBase upper-layer applications.
- You have logged in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose **O&M** > **Backup and Restoration** > **Restoration Management**.

Step 4 Click **Create**.

Step 5 Set **Task Name** to the name of the recovery task.

Step 6 Select the cluster to be operated from **Recovery Object**.

Step 7 In **Restoration Configuration**, select **HBase** under **Service Data**.

Step 8 Set **Path Type** of **HBase** to a backup directory type.

The following backup directory types are supported:

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster. If you select this value, you need to set the following parameters:
  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Maximum number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select **NFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/*.**
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.

  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/*.**
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.

  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**

Step 9 Set **Backup Data** column in **Data Configuration** to one or multiple backup data sources to be recovered. In the **Target Namespace** column, specify the target naming space after backup data recovery.

You are advised to set **Target Namespace** to a location that is difference from the backup naming space.

Step 10 Set **Force recovery** to **YES**, which indicates to forcibly recover all backup data when a data table with the same name already exists. If the data table contains new data added after backup, the new data will be lost after the data recovery. If you set the parameter to **NO**, the recovery task is not executed if a data table with the same name exists.

Step 11 Click **Verify** to check whether the recovery task is configured correctly.

- If the queue name is incorrect, the verification fails.
- If the specified naming space does not exist, the verification fails.
- If the forcibly replacement conditions are not met, the verification fails.

Step 12 Click **OK** to save the settings.

Step 13 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

Step 14 Check whether HBase data is restored in an environment where HBase is newly installed or reinstalled.

- If yes, the administrator needs to set new permission for roles on FusionInsight Manager based on the original service plan.
- If no, the task is complete.

----End

10.11.3.5 Recovering NameNode Data

Scenario

NameNode data needs to be recovered in the following scenarios: HDFS metadata is modified or deleted unexpectedly and needs to be restored. After an administrator performs critical data adjustment in NameNode, an exception occurs or the operation has not achieved the expected result. All modules are faulty and become unavailable. Data is migrated to a new cluster.

System administrators can create a recovery task in FusionInsight Manager to recover NameNode data. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the NameNode data that is generated after the data backup and before the data recovery will be lost.
- It is recommended that a data restoration task restore the metadata of only one component to prevent the data restoration of other components from being affected by stopping a service or instance. If data of multiple components is restored at the same time, data restoration may fail.

  HBase metadata cannot be restored at the same time as NameNode metadata. As a result, data restoration fails.

Impact on the System

- After the data is recovered, the data produced between the backup time and restoration time is lost.
- After the data is recovered, the NameNode needs to be restarted and is unavailable during the restart.
- After data is restored, metadata and service data may not be matched, HDFS enters the security mode, and the HDFS service cannot be started.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- You have logged in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.
- On FusionInsight Manager, all the NameNode role instances whose data is to be recovered are stopped. Other HDFS role instances must keep running. After data is recovered, the NameNode role instances need to be restarted. The NameNode role instances cannot be accessed during the restart.
- The NameNode backup files are stored ***Data path*/LocalBackup/** on the active management node.

Procedure

Step 1 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Instance** > **NameNode** to check whether the NameNode instances of the data to be restored are stopped. If the NameNode instances are not stopped, stop them.

Step 2 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 3 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 4 On FusionInsight Manager, choose **O&M** > **Backup and Restoration** > **Restoration Management**.

Step 5 Click **Create**.

Step 6 Set **Task Name** to the name of the recovery task.

Step 7 Select the cluster to be operated from **Recovery Object**.

Step 8 Select **NameNode**.

Step 9 Set **Path Type** of **NameNode** to a backup directory type.

The settings vary according to backup directory types:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node.

  If you select this value, you need to set the following parameters:

  - **Source Path**: indicates the full path of the local disk for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.

- **RemoteHDFS(DistCp)**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.

  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Queue Name**: indicates the name of the Yarn queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: Enter the IP address of the server where the backup data is stored.

  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
- **OBS**: indicates that the backup files are stored in the OBS directory.

  If you select this value, you need to set the following parameters:
  - **Source Path**: indicates the full path for storing backup data in OBS. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.

    NOTE

    MRS 3.1.0 and later versions support saving backup files to OBS.

Step 10 Click **OK** to save the settings.

Step 11 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.
- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

Step 12 On FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **More** > **Restart Service**.

In the displayed window, enter the password of the current administrator and click **OK**. After the system displays "**Operation succeeded**", click **Finish**. The service is successfully started.

----End

10.11.3.6 Recovering HDFS Service Data

Scenario

HDFS data needs to be recovered in the following scenarios: data is modified or deleted unexpectedly and needs to be restored. After an administrator performs critical data adjustment in HDFS, an exception occurs or the operation has not achieved the expected result. All modules are faulty and become unavailable. Data is migrated to a new cluster.

System administrators can create a recovery task in FusionInsight Manager to recover HDFS data. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the HDFS data that is generated after the data backup and before the data recovery will be lost.
- The HDFS restoration operation cannot be performed for the directories used by running Yarn tasks, for example, **/tmp/logs**, **/tmp/archived**, and **/tmp/hadoop-yarn/staging**. Otherwise, data restoration using Distcp tasks fails due to file loss.

Impact on the System

- During data recovery, user authentication stops and users cannot create new connections.
- After the data is recovered, the data produced between the backup time and restoration time is lost.
- After the data is recovered, the HDFS upper-layer applications need to be started.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- The HDFS backup file save path is correct.
- The HDFS upper-layer applications are stopped.
- You have logged in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Restoration Management**.

Step 4 Click **Create**.

Step 5 Set **Task Name** to the name of the recovery task.

Step 6 Select the cluster to be operated from **Recovery Object**.

Step 7 Select **HDFS**.

Step 8 Set **Path Type** of **HDFS** to a backup directory type.

The following backup directory types are supported:

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.

Step 9 Set **Backup Data** column in **Data Configuration** to one or multiple backup data sources to be recovered based on service requirements. In the **Target Path** column, specify the target location after backup data recovery.

You are advised to set **Target Path** to a new path that is difference from the backup path.

Step 10 Click **Verify** to check whether the recovery task is configured correctly.

- If the queue name is incorrect, the verification fails.
- If the specified directory to be recovered does not exist, the verification fails.

Step 11 Click **OK** to save the settings.

Step 12 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

----End

10.11.3.7 Recovering Hive Service Data

Scenario

Hive data needs to be recovered in the following scenarios: data is modified or deleted unexpectedly and needs to be restored. After an administrator performs

critical data adjustment in Hive, an exception occurs or the operation has not achieved the expected result. All modules are faulty and become unavailable. Data is migrated to a new cluster.

System administrators can create a recovery task in FusionInsight Manager to recover Hive data. Only manual recovery tasks are supported.

Hive backup and recovery cannot identify the service and structure relationships of objects such as Hive tables, indexes, and views. When executing backup and recovery tasks, the user needs to manage a unified recovery point based on the service scenario to ensure proper service running.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover data when the service is running properly, you are advised to manually back up the latest management data before recovering data. Otherwise, the Hive data that is generated after the data backup and before the data recovery will be lost.

Impact on the System

- During data recovery, user authentication stops and users cannot create new connections.
- After the data is recovered, the data produced between the backup time and restoration time is lost.
- After the data is recovered, the Hive upper-layer applications need to be started.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- Plan the database for storing recovered data tables, the HDFS save path of data tables, and the list of users who can access recovered data.
- The Hive backup file save path is correct.
- The Hive upper-layer applications are stopped.
- You have logged in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Restoration Management**.

Step 4 Click **Create**.

Step 5 Set **Task Name** to the name of the recovery task.

Step 6 Select the cluster to be operated from **Recovery Object**.

Step 7 Select **Hive**.

Step 8 Set **Path Type** of **Hive** to a backup directory type.

The following backup directory types are supported:

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.

  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **100**.
- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.

  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
- **SFTP**: Indicates that backup files are stored in the server using SFTP.

  If you select **SFTP**, set the following parameters:
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: Enter the IP address of the server where the backup data is stored.
  - **Port**: Enter the port number used by the SFTP protocol to connect to the backup server. The default value is **22**.
  - **Username**: Enter the username for connecting to the server using SFTP.
  - **Password**: Enter the password for connecting to the server using SFTP.
  - **Source Path**: Enter the full path of the backup file on the backup server. For example, ***backup path/backup task name_data source_task creation time/***.
  - **Queue Name**: indicates the name of the YARN queue used for backup task execution.
  - **Recovery Point List**: Click **Refresh** and select an HDFS directory that has been backed up in the standby cluster.
  - **Target NameService Name**: indicates the target NameService that corresponds to the selected backup directory. The default value of this parameter is **hacluster**.
  - **Maximum Number of Maps**: indicates the maximum number of maps in a MapReduce task. The default value of this parameter is **20**.
  - **Maximum Bandwidth of a Map (MB/s)**: indicates the maximum bandwidth of a map. The default value of this parameter is **1**.

Step 9 Set **Backup Data** in the **Data Configuration** to one or multiple backup data sources to be recovered based on service requirements. In the **Target Database** and **Target Path** columns, specify the target database and file save path after backup data recovery.

Configuration restrictions:

- Data can be recovered to the original database, but data tables must be stored in a new path that is difference from the backup path.
- To recover Hive index tables, select the Hive data tables that correspond to the Hive index tables to be recovered.
- If a new recovery directory is selected to avoid affecting the current data, HDFS permission must be manually granted so that users who have permission of backup tables can access this directory.
- Data can be recovered to other databases. In this case, HDFS permission must be manually granted so that users who have permission of backup tables can access the HDFS directory that corresponds to the database.

Step 10 Set **Force recovery** to **YES**, which indicates to forcibly recover all backup data when a data table with the same name already exists. If the data table contains new data added after backup, the new data will be lost after the data recovery. If you set the parameter to **NO**, the recovery task is not executed if a data table with the same name exists.

Step 11 Click **Verify** to check whether the recovery task is configured correctly.

- If the queue name is incorrect, the verification fails.
- If the specified directory to be recovered does not exist, the verification fails.
- If the forcibly replacement conditions are not met, the verification fails.

Step 12 Click **OK** to save the settings.

Step 13 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

----End

10.11.3.8 Recovering Kafka Metadata

Scenario

Kafka metadata needs to be recovered in the following scenarios: Data is modified or deleted unexpectedly and needs to be restored; after an administrator performs a critical operation (such as upgrade and critical data adjustment) on ZooKeeper, an exception occurs or the operation has not achieved the expected result; all Kafka modules are faulty and become unavailable; data is migrated to a new cluster.

System administrators can create a recovery task in FusionInsight Manager to recover Kafka data. Only manual recovery tasks are supported.

NOTICE

- Data recovery can be performed only when the system version is consistent with that of data backup.
- To recover Kafka metadata when the service is running properly, you are advised to manually back up the latest Kafka metadata before recovery. Otherwise, the Kafka metadata that is generated after the data backup and before the data recovery will be lost.

Impact on the System

- After the metadata is recovered, the data generated between the backup point in time and the recovery point in time is lost.

- After the metadata is recovered, the offset information stored on ZooKeeper by Kafka consumers is restored to a previous state, resulting in repeated consumption.

Prerequisites

- To restore data from the remote HDFS, you need to prepare the standby cluster. If the active cluster is deployed in security mode and the active and standby clusters are not managed by the same FusionInsight Manager, configure system mutual trust. For details, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**. If the active cluster is deployed in normal mode, do not configure mutual trust.
- Cross-cluster replication has been configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Replication**.
- The time of the active cluster and standby cluster must be the same, and the NTP service in the active and standby clusters must use the same time source.
- The Kafka service is disabled first, and then enabled upon data recovery.
- You have logged in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration > Backup Management**.

Step 2 In the **Operation** column of a specified task in the task list, click **More > View History** to view historical backup task execution records.

In the displayed window, locate a specified success record and click **View** in the **Backup Path** column to view the backup path information of the task and find the following information:

- **Backup Object** specifies the data source of the backup data.
- **Backup Path** specifies the full path where the backup files are saved.

  Select the correct item, and manually copy the full path of backup files in **Backup Path**.

Step 3 On FusionInsight Manager, choose **O&M > Backup and Restoration > Restoration Management**.

Step 4 Click **Create**.

Step 5 Set **Task Name** to the name of the recovery task.

Step 6 Select the cluster to be operated from **Recovery Object**.

Step 7 Select **Kafka** in **Restoration Configuration**.

NOTE

If there are multiple Kafka services, you can specify the Kafka to be recovered.

Step 8 Set **Path Type** of **Kafka** to a backup directory type.

The settings vary according to backup directory types:

- **LocalDir**: indicates that the backup files are stored on the local disk of the active management node.

  If you select this value, you need to set **Source Path** to select the backup file, for example, ***version_data source_task execution time.tar.gz***.

- **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster.

  If you select this value, you need to set the following parameters:

  - **Source Path**: indicates the full path of the backup file in HDFS, for example, ***backup path/backup task name_task creation time/version_data source_task execution time.tar.gz***.
  - **Source NameService Name**: indicates the NameService name that corresponds to the backup directory when the recovery task is executed. The default value is **hacluster**.

- **RemoteHDFS**: indicates that the backup files are stored in the HDFS directory of the standby cluster.

  If you select this value, you need to set the following parameters:

  - **Source NameService Name**: indicates the NameService name of the backup data cluster. You can set it to the NameService name (**haclusterX**, **haclusterX1**, **haclusterX2**, **haclusterX3**, or **haclusterX4**) of the built-in remote cluster of the cluster, or the NameService name of a configured remote cluster.
  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Source NameNode IP Address**: indicates the NameNode service plane IP address of the standby cluster, supporting the active node or standby node.
  - **Source Path**: indicates the full path of the HDFS directory for storing standby cluster backup data. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.
  - **Queue Name**: indicates the name of the Yarn queue used for backup task execution. The name must be the same as the name of the queue that is running properly in the cluster.

- **NFS**: indicates that backup files are stored in the NAS over the NFS protocol.

  If you select NFS, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**

- **CIFS**: indicates that backup files are stored in the NAS over the CIFS protocol.

  If you select **CIFS**, set the following parameters:

  - **IP Mode**: mode of the target IP address.The system automatically selects the IP address mode based on the cluster network type, for example, **IPv4** or **IPv6**.
  - **Server IP address**: indicates the NAS server IP address.
  - **Port**: indicates the port ID used by the CIFS protocol to connect to the NAS server. The default value is **445**.
  - **Username**: Indicates the user name that is configured when setting the CIFS protocol.
  - **Password**: indicates the password that is configured when setting the CIFS protocol.
  - **Source Path**: indicates the complete path of the backup file on the NAS server. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time*.tar.gz.**
- **OBS**: indicates that the backup files are stored in the OBS directory.

  If you select this value, you need to set the following parameters:

  - **Source Path**: indicates the full path for storing backup data in OBS. For example, ***backup path/backup task name_data source_task creation time/version_data source_task execution time.tar.gz***.

    NOTE

    MRS 3.1.0 and later versions support saving backup files to OBS.

Step 9 Click **OK** to save the settings.

Step 10 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

- After the recovery is successful, the progress bar is in green.
- After the recovery is successful, the recovery task cannot be executed again.
- If the recovery task fails during the first execution, rectify the fault and click **Retry** to execute the task again.

----End

10.11.4 Enabling Cross-Cluster Replication

Scenario

DistCp is used to replicate the data stored in HDFS from a cluster to another cluster. DistCp depends on the cross-cluster replication function, which is disabled by default. You need to enable it for both clusters.

This section describes how to modify parameters on FusionInsight Manager to enable the cross-cluster replication function. After this function is enabled, you can create a backup task for backing up data to the remote HDFS (RemoteHDFS).

Impact on the System

Yarn needs to be restarted to enable the cross-cluster replication function and cannot be accessed during restart.

Prerequisites

- The **hadoop.rpc.protection** parameter of HDFS in the two clusters for data replication must use the same data transmission mode. The default value is **privacy**, indicating encrypted transmission. The value **authentication** indicates that transmission is not encrypted.
- For clusters in security mode, you need to configure mutual trust between clusters.

Procedure

Step 1 Log in to FusionInsight Manager of one of the two clusters.

Step 2 Choose **Cluster** > ***Name of the desired cluster*** **> Services** > **Yarn** > **Configurations**, and click **All Configurations**.

Step 3 In the navigation pane, choose **Yarn** > **Distcp**.

Step 4 Modify **dfs.namenode.rpc-address**, set **haclusterX.remotenn1** to the service IP address and RPC port of one NameNode instance of the peer cluster, and set **haclusterX.remotenn2** to the service IP address and RPC port number of the other NameNode instance of the peer cluster.

haclusterX.remotenn1 and **haclusterX.remotenn2** do not distinguish active and standby NameNodes. The default NameNode RPC port is 8020 and cannot be modified on Manager.

Examples of modified parameter values: **10.1.1.1:8020** and **10.1.1.2:8020**.

NOTE

- If data of the current cluster needs to be backed up to the HDFS of multiple clusters, you can configure the corresponding NameNode RPC addresses to haclusterX1, haclusterX2, haclusterX3, and haclusterX4.

Step 5 Click **Save**. In the confirmation dialog box, click **OK**.

Step 6 Restart the Yarn service.

Step 7 Log in to FusionInsight Manager of the other cluster and repeat **Step 2** to **Step 6**.

----End

10.11.5 Managing Local Quick Recovery Tasks

Scenario

When DistCp is used to back up data, the backup snapshot is saved to HDFS of the active cluster. FusionInsight Manager supports using the local snapshot for quick data recovery, requiring less time than recovering data from the standby cluster.

Use FusionInsight Manager and the snapshots on HDFS of the active cluster to create a local quick recovery task and execute the task.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration** > **Backup Management**.

Step 2 In the backup task list, locate a created task and click **Restore** in the **Operation** column.

Step 3 Check whether the system displays **No data is available for quick restoration. Create a task on the restoration management page to restore data**.

- If yes, click **OK** to close the dialog box. No backup data snapshot is created in the active cluster, and no further action is required.
- If no, go to **Step 4** to create a local quick recovery task.

  NOTE

  Metadata does not support quick restoration.

Step 4 Set **Name** to the name of the local quick recovery task.

Step 5 Set **Configuration** to a data source.

Step 6 Set **Recovery Point List** to a recovery point that contains the backup data.

Step 7 Set **Queue Name** to the name of the Yarn queue used in the task execution. The name must be the same as the name of the queue that is running properly in the cluster.

Step 8 Set **Data Configuration** to the object to be recovered.

Step 9 Click **Verify**. After "The restoration task configuration is verified successfully." is displayed, click **OK**.

Step 10 Click **OK**.

Step 11 In the recovery task list, locate a created task and click **Start** in the **Operation** column to execute the recovery task.

After the task is complete, **Task Status** of the task is displayed as **Successful**.

----End

10.11.6 Modifying a Backup Task

Scenario

Modify the parameters of a created backup task on FusionInsight Manager to meet changing service requirements. The parameters of recovery tasks can only be viewed but cannot be modified.

Impact on the System

After a backup task is modified, the new parameters take effect when the task is executed next time.

Prerequisites

- A backup task has been created.
- A new backup task policy has been planned based on the actual situation.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Backup and Restoration** > **Backup Management**.

Step 2 In the task list, locate a specified task, click **Configure** in the **Operation** column to go to the configuration modification page.

On the displayed page, modify the following parameters:

- **Started**
- **Period**
- **Destination NameService Name**
- **Target NameNode IP Address**
- **Target Path**
- **Maximum Number of Backup Copies**
- **Maximum Number of Recovery Points**
- **Maximum Number of Maps**
- **Maximum Bandwidth of a Map**

NOTE

After the **Target Path** parameter of a backup task is modified, this task will be performed as a full backup task for the first time by default.

Step 3 Click **OK** to save the settings.

----End

10.11.7 Viewing Backup and Recovery Tasks

Scenario

On FusionInsight Manager, view created backup and recovery tasks and check their running status.

Prerequisites

You have logged in to FusionInsight Manager. For details, see Logging In to the Management System.

Procedure

Step 1 On FusionInsight Manager, choose **O&M > Backup and Restoration**.

Step 2 Click **Backup Management** or **Restoration Management**.

Step 3 In the task list, obtain the previous execution result in the **Task Status** and **Task Progress** column. Green indicates that the task is executed successfully, and red indicates that the execution fails.

Step 4 In the task list, locate a specified task and choose **More > View History** or **click View History** in the **Operation** column to display historical records of backup and recovery task execution.

In the displayed window, click before a specified record to display log information about the execution.

----End

Related Tasks

- Starting Backup and Recovery Tasks

  In the task list, locate a specified task and choose **More** > **Back Up Now** or click **Start** in the **Operation** column to start a backup or recovery task that is ready or fails to be executed. Executed recovery tasks cannot be repeatedly executed.

- Stopping Backup and Recovery Tasks

  In the task list, locate a specified task and choose **More** > **Stop** or click **Stop** in the **Operation** column to start a backup or recovery task that is running. After the task is successfully stopped, its **Task Status** changes to **Stopped**.

- Deleting Backup and Recovery Tasks

  In the task list, locate a specified task and choose **More** > **Delete** or click **Delete** in the **Operation** column to delete a backup or recovery task. Backup data will be reserved by default after a task is deleted.

- Suspending Backup Tasks

  In the task list, locate a specified task and choose **More** > **Suspend** in the **Operation** column to suspend a backup task. Only periodic backup tasks can be suspended. Suspended backup tasks are no longer executed automatically. When you suspend a backup task that is being executed, the task execution stops. To Resume a task, choose **More** > **Resume**.

10.12 Security Management

10.12.1 Security Overview

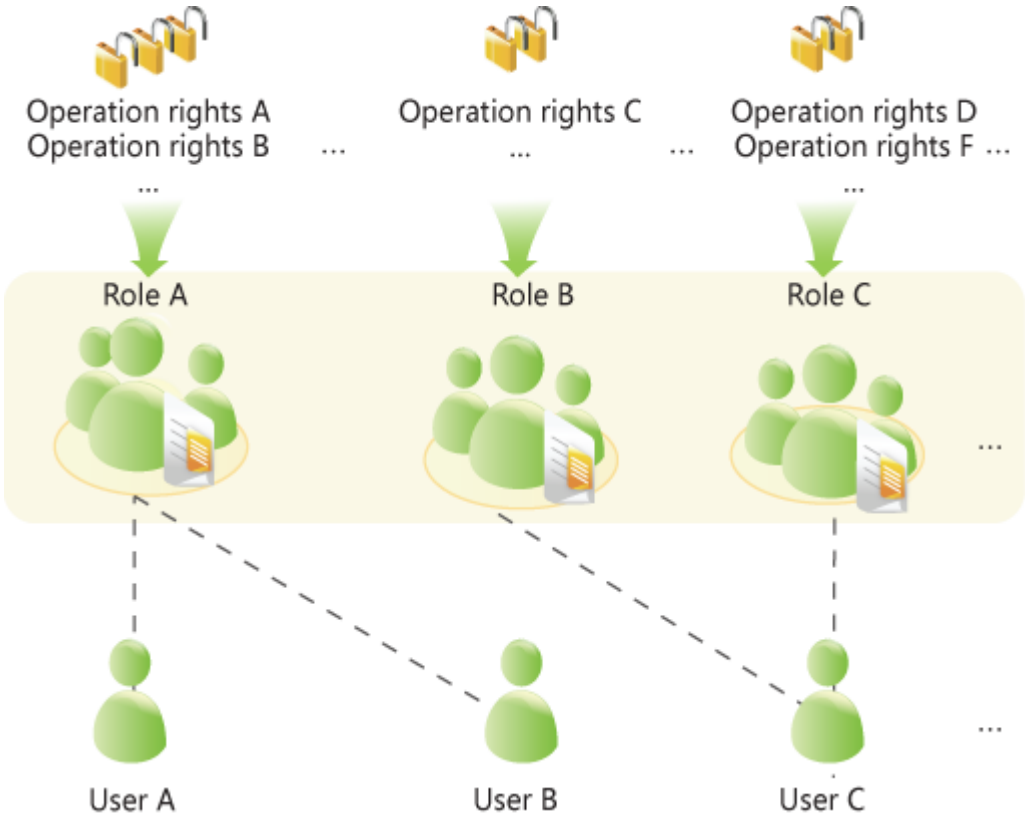
10.12.1.1 Rights Model

Role-based Access Control

FusionInsight adopts the role-based access control (RBAC) mode to perform rights management on the big data system. It integrates the rights management functions of the components to centrally manage rights. Common users are shielded from internal rights management details, and administrators' rights management operations are simplified, improving rights management usability and user experience.

The rights model of FusionInsight is "users-user groups-roles-rights".

Figure 10-21 Rights model



- **Rights**

  Rights are defined by components and allow users to access resources of components. Different components have different rights for their resources.

  Example:

  – HDFS provides read, write, and execute permissions on file resources.

  – HBase provides create, write, and read permissions on table resources.

- **Role**

  Role is a collection of component rights. Each role can have multiple rights of multiple components. Different roles can have the rights of a resource of one component.

- **User group**

  User group is a collection of users. When a user group is bound to a role, users in this group obtain the rights defined by the role.

  Different user groups can be associated with the same role, and a user group can be associated with no role. In principle, the user group does not have the rights of any component resources.

  NOTE

  In some components, the system grants related rights to specific user groups by default.

- **User**

  Users are visitors to the system. Each user has the rights of the user group and role associated with the user. Users need to be added to the user group or associated roles to obtain the corresponding rights.

Policy-based Access Control

The Ranger component uses policy-based access control (PBAC) to manage permissions and implement fine-grained data access control on components such as HDFS, Hive, and HBase.

NOTE

The component supports only one permission control mechanism. After the Ranger permission control policy is enabled for the component, the permission on the component in the role created on FusionInsight Manager becomes invalid (The ACL rules of HDFS and Yarn components still take effect). You need to add a policy on the Ranger management page to grant permissions on resources.

The ranger permission model consists of multiple permission policies. The permission policies are as follows:

- **Resource**

  Objects provided by components for users to access, such as HDFS files or folders, queues in Yarn, and databases, tables, and columns in Hive.

- **User**

  Indicates the user who accesses the system. The rights of each user are obtained based on the policy associated with the user. Information about users, user groups, and roles in the LDAP is periodically synchronized to the Ranger.

- **Permission**

  In a policy, you can configure various access conditions for resources, such as file read/write, permission conditions, rejection conditions, and exception conditions.

10.12.1.2 Rights Mechanism

FusionInsight adopts the Lightweight Directory Access Protocol (LDAP) to store data of users and user groups. Information about role definitions is stored in the relational database and the mapping between roles and rights is saved in components.

FusionInsight uses Kerberos for unified authentication.

The verification process of user rights is as follows:

1. A client (a user terminal or FusionInsight component service) invokes the FusionInsight authentication interface.
2. FusionInsight uses the login username and password for Kerberos authentication.
3. If the authentication succeeds, the client sends a request for accessing the server (a FusionInsight component service).
4. The server finds the user group and role to which the login user belongs.

5. The server obtains all rights of the user group and the role.
6. The server determines whether the client has the permission to access the resources it applies for.

Example (RBAC):

There are three files in HDFS, fileA, fileB, and fileC.

- roleA has read and write permissions for fileA and roleB has the read permission for fileB.
- groupA is bound to roleA and groupB is bound to roleB.
- userA belongs to groupA and roleB, and userB belongs to groupB.

When userA successfully logs in to the system and accesses HDFS:

1. HDFS obtains the role (roleB) to which userA is bound.
2. HDFS also obtains the role (roleA) to which the user group of userA is bound.
3. In this case, userA has all the rights of roleA and roleB.
4. As a result, userA has read and write permissions for fileA, has the read permission on fileB, and has no permission for fileC.

Similarly, when userB successfully logs in to the system and accesses HDFS:

1. userB only has the rights of roleB.
2. As a result, userB has the read permission on fileB, and has no permissions for fileA and fileC.

10.12.1.3 Authentication Policies

The big data platform performs user identity authentication to prevent invalid users from accessing the cluster. The cluster provides authentication capabilities in both Security Mode and Normal mode.

Security Mode

The cluster in Security Mode uses the Kerberos authentication protocol to perform security authentication. The Kerberos protocol supports mutual authentication between the client and the server. This improves security and eliminates the security risks causes by using the network to send user credentials to simulate authentication. In cluster, KrbServer service provides Kerberos authentication support.

Kerberos user object

In the Kerberos protocol, a user object is a principal. A complete user object consists of a username and domain name. In O&M management or application development scenarios, a user can connect to the cluster server only after the user is authenticated on the client. In O&M and service scenarios, **Human-machine** and **Machine-machine** users are used. The difference between **Human-machine** and **Machine-machine** users is that the passwords of **Machine-machine** users are randomly generated by the system.

Kerberos authentication

The Kerberos authentication supports two modes: password authentication mode and keytab authentication mode. The validity period of authentication is 24 hours by default.

- Password authentication: Identity authentication is performed by entering the correct password of a user. This mode is mainly used in O&M management scenarios where **Human-machine** users are used. The command is **kinit** *Username*.
- Keytab authentication: The keytab file includes the user principal and encryption information of user credentials. When the keytab file is used for authentication, the system automatically uses encrypted credential information to perform authentication and the user password does not need to be entered. This mode is mainly used in component application development scenarios where **Machine-machine** users are used. The keytab file can also be used in the **kinit** command.

Normal Mode

When the cluster is in Normal Mode, different components use different open-source authentication mechanisms, and the **kinit** authentication command is not supported. FusionInsight Manager (including DBService, KrbServer, and LdapServer) uses the username and password authentication mode. **Table 10-80** lists the authentication mechanisms used by components.

Table 10-80 Component authentication modes

Service	Authentication Mode
CDL	No authentication
ClickHouse	Simple authentication
Flume	No authentication
HBase	• WebUI: No authentication • Client: Simple authentication
HDFS	• WebUI: No authentication • Client: Simple authentication
Hive	Simple authentication
Hue	Username and password authentication
Kafka	No authentication
Loader	• WebUI: Username and password authentication • Client: No authentication
Mapreduce	• WebUI: No authentication • Client: No authentication
Oozie	• WebUI: Username and password authentication • Client: Simple authentication

Service	Authentication Mode
Spark2x	• WebUI: No authentication • Client: Simple authentication
Storm	No authentication
Yarn	• WebUI: No authentication • Client: Simple authentication
ZooKeeper	Simple authentication

The authentication modes are described as follows:

- **Simple authentication**: During the connection from the client to the server, the execution user on the client (such as the OS user **root** or **omm**) is used for automatic authentication by default. Administrators or service users are unaware of the authentication and do not need to run the **kinit** command to perform the authentication.
- **Username and password authentication**: The usernames and passwords of **Human-machine** users are used for authentication.
- **No authentication**: Any user can access the server by default.

10.12.1.4 Permission Verification Policies

Security Mode

After a user is authenticated by the big data platform, the system determines whether to verify the user's permission based on the actual permission management configuration to ensure that the user has limited or all permission on resources. If the user does not have sufficient permission, the user can access resources only after the system administrator grant related permission on each component to the user. The cluster provides permission verification capabilities in both Security Mode and Normal Mode. Permission on components is the same in the two modes.

By default, the Ranger service is installed and Ranger authentication is enabled for a newly installed cluster in security mode. You can set fine-grained security access policies for accessing component resources through the permission plug-in of the component. If Ranger authentication is not required, the administrator can manually disable Ranger authentication on the service page. After Ranger authentication is disabled, the system continues to perform permission control based on the role model of FusionInsight Manager when accessing component resources.

In a cluster in security mode, the following components support Ranger authentication: HDFS, Yarn, Kafka, Hive, HBase, Storm, Spark2x, Impala.

In a cluster upgraded from an earlier version, Ranger authentication is not used by default when users access component resources. The administrator can manually enable Ranger authentication after installing the Ranger service.

By default, all components in the cluster in Security Mode perform permission verification on access in a unified manner, and the permission verification function cannot be disabled.

Normal Mode

Different components in the cluster in Normal Mode use different open-source permission verification behavior. **Table 10-81** lists detailed permission verification mechanisms.

In a cluster in non-security mode, the Ranger supports permission control on component resources based on OS users. The following components support Ranger authentication: HBase, HDFS, Hive, Spark2x, and Yarn.

Table 10-81 Component permission verification modes in Normal Mode

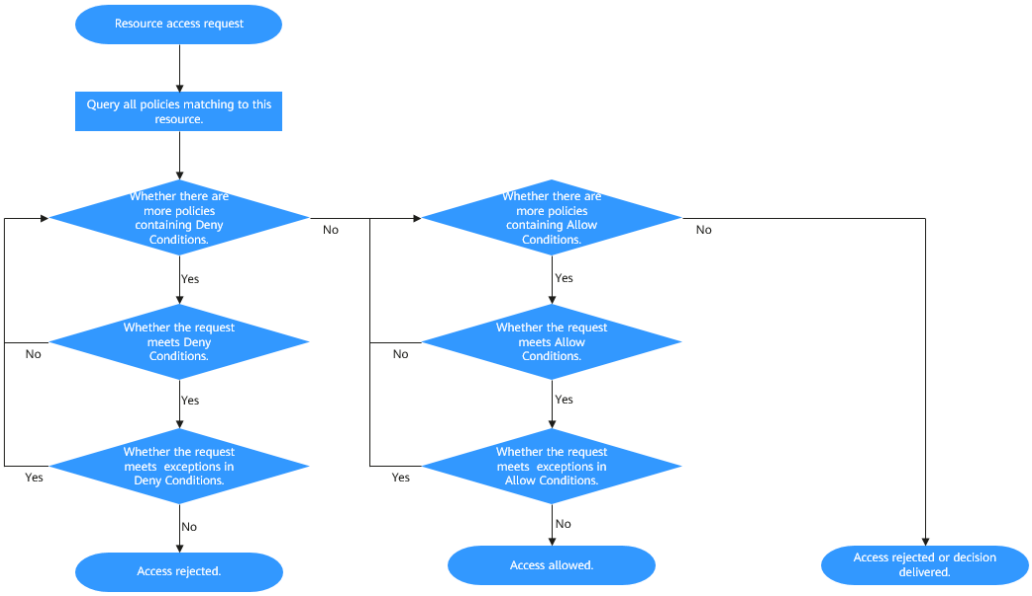
Service	Permission Verification	Permission Verification Enabling and Disabling
ClickHouse	Required	Not supported
Flume	Not required	Not supported
HBase	Not required	Supported
HDFS	Required	Supported
Hive	Not required	Not supported
Hue	Not required	Not supported
Kafka	Not required	Not supported
Loader	Not required	Not supported
Mapreduce	Not required	Not supported
Oozie	Required	Not supported
Spark2x	Not required	Not supported
Storm	Not required	Not supported
Yarn	Not required	Supported
ZooKeeper	Required	Supported

Condition Priorities of the Ranger Permission Policy

When configuring a permission policy for a resource, you can configure Allow Conditions, Exclude from Allow Conditions, Deny Conditions, and Exclude from Deny Conditions for the resource, to meet unexpected requirements in different scenarios.

The priorities of different conditions are listed in descending order: Exclude from Deny Conditions > Deny Conditions > Exclude from Allow Conditions > Allow Conditions

The following figure shows the process of determining condition priorities. If the component resource request does not match the permission policy in Ranger, the system rejects the access by default. However, for HDFS and Yarn, the system delivers the decision to the access control layer of the component for determination.



For example, if you want to grant the read and write permissions of the **FileA** folder to the **groupA** user group, but the user in the group is not **UserA**, you can add an allowed condition and an exception condition.

10.12.1.5 User Account List

User Classification

The MRS cluster provides the following three types of users. The system administrator needs to periodically change the passwords. It is not recommended to use the default passwords.

NOTE

This section describes the default users in the MRS cluster.

User Type	Description
System users	• User created on FusionInsight Manager for O&M and service scenarios. There are two types of users: – **Human-machine** user: used in scenarios such as FusionInsight Manager O&M and operations on a component client. When creating a user of this type, you need to set password and confirm password by referring to **Creating a User**. – **Machine-machine** user: used for system application development. • User who runs OMS processes

User Type	Description
Internal system users	Internal user to perform Kerberos authentication, process communications, save user group information, and associate user permissions. It is recommended that internal system users not be used in O&M scenarios. Operations can be performed as user **admin** or another user created by the system administrator based on service requirements.
Database users	• User who manages OMS database and accesses data • User who runs service components (Hue, Hive, Loader, Oozie, Ranger, and DBService) in the database.

System Users

NOTE

- User **root** of the OS is required, the password of user **root** on all nodes must be the same.
- User **ldap** of the OS is required. Do not delete this account. Otherwise, the cluster may not work properly. The OS administrator maintains the password management policies.

User Type	Username	Initial Password	Description	Password Change Method
System administrator	admin	User-defined password	FusionInsight Manager administrator. **NOTE** By default, user **admin** does not have the management permission on other components. For example, when accessing the native UI of a component, the user fails to access the complete component information due to insufficient management permission on the component.	For details, see **Changing the Password for User admin**.

User Type	Username	Initial Password	Description	Password Change Method
Node OS user	ommdba	Random password	User that creates the system database. This user is an OS user generated on the management node and does not require a unified password. This account cannot be used for remote login.	For details, see **Changing the Password for an OS User**.
	omm	Bigdata123@	Internal running user of the system. This user is an OS user generated on all nodes and does not require a unified password.	

Internal System Users

User Type	Default User	Initial Password	Description	Password Change Method
Kerberos administrator	kadmin/admin	Admin@123	Used to add, delete, modify, and query user accounts on Kerberos.	For details, see **Changing the Password for the Kerberos Administrator**.
OMS Kerberos administrator	kadmin/admin	Admin@123	Used to add, delete, modify, and query user accounts on OMS Kerberos.	For details, see **Changing the Password for the OMS Kerberos Administrator**.
LDAP administrator	cn=root,dc=hadoop,dc=com	LdapChangeMe@123	Used to add, delete, modify, and query the user account information on LDAP.	For details, see **Changing the Passwords of the LDAP Administrator and the LDAP User (Including OMS LDAP)**.

User Type	Default User	Initial Password	Description	Password Change Method
OMS LDAP administrator	cn=root,dc=hadoop,dc=com	LdapChangeMe@123	Used to add, delete, modify, and query the user account information on OMS LDAP.	
LDAP user	cn=pg_search_dn,ou=Users,dc=hadoop,dc=com	Randomly generated by the system	Used to query information about users and user groups on LDAP.	
OMS LDAP user	cn=pg_search_dn,ou=Users,dc=hadoop,dc=com	Randomly generated by the system	Used to query information about users and user groups on OMS LDAP.	
LDAP administrator account	cn=krbkdc,ou=Users,dc=hadoop,dc=com	LdapChangeMe@123	Used to query Kerberos component authentication account information.	For details, see **Changing the Password for the LDAP Administrator**.
	cn=krbadmin,ou=Users,dc=hadoop,dc=com	LdapChangeMe@123	Used to add, delete, modify, and query Kerberos component authentication account information.	

User Type	Default User	Initial Password	Description	Password Change Method
Component running user	hdfs	Hdfs@123	This user is the HDFS system administrator and has the following permissions: 1. File system operation permissions: • Views, modifies, and creates files. • Views and creates directories. • Views and modifies the groups where files belong. • Views and sets disk quotas for users. 2. HDFS management operation permissions: • Views the web UI status. • Views and sets the active and standby HDFS status. • Enters and exits the HDFS in security mode. • Checks the HDFS file system. 3. Logs in to the FTP service page.	For details, see **Changing the Password for a Component Running User**.

User Type	Default User	Initial Password	Description	Password Change Method
	hbase	Hbase@123	This user is the HBase and HBase1 to HBase4 system administrator and has the following permissions: ● Cluster management permission: Performs **Enable** and **Disable** operations on tables to trigger MajorCompact and ACL operations. ● Grants and revokes permissions, and shuts down the cluster. ● Table management permission: Creates, modifies, and deletes tables. ● Data management permission: Reads data in tables, column families, and columns. ● Logs in to the HMaster web UI. ● Logs in to the FTP service page.	

User Type	Default User	Initial Password	Description	Password Change Method
	mapred	Mapred@123	This user is the MapReduce system administrator and has the following permissions: ● Submits, stops, and views the MapReduce tasks. ● Modifies the Yarn configuration parameters. ● Logs in to the FTP service page. ● Logs in to the Yarn web UI.	
	zookeeper	ZooKeeper@123	This user is the ZooKeeper system administrator and has the following permissions: ● Adds, deletes, modifies, and queries all nodes in ZooKeeper. ● Modifies and queries quotas of all nodes in ZooKeeper.	
	rangeradmin	Rangeradmin@123	This user has the Ranger system management permissions and user permissions: ● Ranger web UI management permission ● Management permission of each component that uses Ranger authentication	
	rangeraudit	Rangerauditor@123	Default audit user of the Ranger system.	

User Type	Default User	Initial Password	Description	Password Change Method
	hive	Hive@123	This user is the Hive system administrator and has the following permissions: 1. Hive administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks. 4. Ranger policy management permission	

User Type	Default User	Initial Password	Description	Password Change Method
	hive1	Hive1@123	This user is the Hive1 system administrator and has the following permissions: 1. Hive1 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks. 4. Ranger policy management permission	

User Type	Default User	Initial Password	Description	Password Change Method
	hive2	Hive2@123	This user is the Hive2 system administrator and has the following permissions: 1. Hive2 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks. 4. Ranger policy management permission	

User Type	Default User	Initial Password	Description	Password Change Method
	hive3	Hive3@123	This user is the Hive3 system administrator and has the following permissions: 1. Hive3 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks. 4. Ranger policy management permission	

User Type	Default User	Initial Password	Description	Password Change Method
	hive4	Hive4@123	This user is the Hive4 system administrator and has the following permissions: 1. Hive4 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks. 4. Ranger policy management permission	

User Type	Default User	Initial Password	Description	Password Change Method
	kafka	Kafka@123	This user is the Kafka system administrator and has the following permissions: ● Creates, deletes, produces, and consumes the topic; modifies the topic configuration. ● Controls the cluster metadata, modifies the configuration, migrates the replica, elects the leader, and manages ACL. ● Submits, queries, and deletes the consumer group offset. ● Queries the delegation token. ● Queries and submits the transaction.	
	storm	Admin@123	Storm system administrator User permission: Submits Storm tasks.	
	rangerusersync	Randomly generated by the system	Synchronizes users and internal users of user groups.	
	rangertagsync	Randomly generated by the system	Internal user for synchronizing tags.	

User Type	Default User	Initial Password	Description	Password Change Method
	oms/ manager	Randomly generated by the system	Controller and NodeAgent authentication user. The user has the permission on the **supergroup** group.	
	backup/ manager	Randomly generated by the system	User for running backup and restoration tasks. The user has the permission on the **supergroup**, **wheel**, and **ficommon** groups. After cross-system mutual trust is configured, the user has the permission to access data in the HDFS, HBase, Hive, and ZooKeeper systems.	

User Type	Default User	Initial Password	Description	Password Change Method
	hdfs/ hadoop.*<System domain name>*	Randomly generated by the system	This user is used to start the HDFS and has the following permissions: 1. File system operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. ● Views and sets disk quotas for users. 2. HDFS management operation permissions: ● Views the web UI status. ● Views and sets the active and standby HDFS status. ● Enters and exits the HDFS in security mode. ● Checks the HDFS file system. 3. Logs in to the FTP service page.	

User Type	Default User	Initial Password	Description	Password Change Method
	mapred/hadoop.*<System domain name>*	Randomly generated by the system	This user is used to start the MapReduce and has the following permissions: ● Submits, stops, and views the MapReduce tasks. ● Modifies the Yarn configuration parameters. ● Logs in to the FTP service page. ● Logs in to the Yarn web UI.	
	mr_zk/hadoop.*<System domain name>*	Randomly generated by the system	Used for MapReduce to access ZooKeeper.	
	hbase/hadoop.*<System domain name>*	Randomly generated by the system	User for the authentication between internal components during the HBase system startup.	
	hbase/zkclient.*<System domain name>*	Randomly generated by the system	User for HBase to perform ZooKeeper authentication in a security mode cluster.	
	thrift/hadoop.*<System domain name>*	Randomly generated by the system	ThriftServer system startup user.	

User Type	Default User	Initial Password	Description	Password Change Method
	thrift/*<hostname>*	Randomly generated by the system	User for the ThriftServer system to access HBase. This user has the read, write, execution, creation, and administration permission on all NameSpaces and tables of HBase. *<hostname>* indicates the name of the host where the ThriftServer node is installed in the cluster.	

User Type	Default User	Initial Password	Description	Password Change Method
	hive/hadoop.*<System domain name>*	Randomly generated by the system	User for the authentication between internal components during the Hive system startup. The user permissions are as follows: 1. Hive administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks.	

User Type	Default User	Initial Password	Description	Password Change Method
	hive1/ hadoop.*<System domain name>*	Randomly generated by the system	User for the authentication between internal components during the Hive1 system startup. The user permissions are as follows: 1. Hive1 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks.	

User Type	Default User	Initial Password	Description	Password Change Method
	hive2/hadoop.*<System domain name>*	Randomly generated by the system	User for the authentication between internal components during the Hive2 system startup. The user permissions are as follows: 1. Hive2 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks.	

User Type	Default User	Initial Password	Description	Password Change Method
	hive3/hadoop.*<System domain name>*	Randomly generated by the system	User for the authentication between internal components during the Hive3 system startup. The user permissions are as follows: 1. Hive3 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks.	

User Type	Default User	Initial Password	Description	Password Change Method
	hive4/hadoop.*<System domain name>*	Randomly generated by the system	User for the authentication between internal components during the Hive4 system startup. The user permissions are as follows: 1. Hive4 administrator permissions: ● Creates, deletes, and modifies a database. ● Creates, queries, modifies, and deletes a table. ● Queries, inserts, and uploads data. 2. HDFS file operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. 3. Submits and stops the MapReduce tasks.	
	loader/hadoop.*<System domain name>*	Randomly generated by the system	User for Loader system startup and Kerberos authentication	

User Type	Default User	Initial Password	Description	Password Change Method
	HTTP/*<hostname>*	Randomly generated by the system	Used to connect to the HTTP interface of each component. *<hostname>* indicates the host name of a node in the cluster.	
	hue	Randomly generated by the system	User for Hue system startup, Kerberos authentication, and HDFS and Hive access	
	flume	Randomly generated by the system	User for Flume system startup and HDFS and Kafka access. The user has read and write permission of the HDFS directory **/flume**.	
	flume_server	Randomly generated by the system	User for Flume system startup and HDFS and Kafka access. The user has read and write permission of the HDFS directory **/flume**.	
	spark2x/hadoop.*<System domain name>*	Randomly generated by the system	This user is the Spark2x system administrator and has the following user permissions: 1. Starts the Spark2x service. 2. Submits Spark2x tasks.	
	spark_zk/hadoop.*<System domain name>*	Randomly generated by the system	Used for Spark2x to access ZooKeeper.	

User Type	Default User	Initial Password	Description	Password Change Method
	spark2x1/hadoop.*<System domain name>*	Randomly generated by the system	This user is the Spark2x1 system administrator and has the following user permissions: 1. Starts the Spark2x1 service. 2. Submits Spark2x tasks.	
	spark2x2/hadoop.*<System domain name>*	Randomly generated by the system	This user is the Spark2x2 system administrator and has the following user permissions: 1. Starts the Spark2x2 service. 2. Submits Spark2x tasks.	
	spark2x3/hadoop.*<System domain name>*	Randomly generated by the system	This user is the Spark2x3 system administrator and has the following user permissions: 1. Starts the Spark2x3 service. 2. Submits Spark2x tasks.	
	spark2x4/hadoop.*<System domain name>*	Randomly generated by the system	This user is the Spark2x4 system administrator and has the following user permissions: 1. Starts the Spark2x4 service. 2. Submits Spark2x tasks.	
	zookeeper/hadoop.*<System domain name>*	Randomly generated by the system	ZooKeeper system startup user.	

User Type	Default User	Initial Password	Description	Password Change Method
	zkcli/hadoop.*<System domain name>*	Randomly generated by the system	ZooKeeper server login user.	
	oozie	Randomly generated by the system	User for Oozie system startup and Kerberos authentication.	
	kafka/hadoop.*<System domain name>*	Randomly generated by the system	Used for security authentication of Kafka.	
	storm/hadoop.*<System domain name>*	Randomly generated by the system	Storm system startup user.	
	storm_zk/hadoop.*<System domain name>*	Randomly generated by the system	Used for the Worker process to access ZooKeeper.	
	flink/hadoop.*<System domain name>*	Randomly generated by the system	Internal user of the Flink service.	
	check_ker_M	Randomly generated by the system	User who performs a system internal test about whether the Kerberos service is normal.	
	tez	Randomly generated by the system	User for TezUI system startup, Kerberos authentication, and access to Yarn	

User Type	Default User	Initial Password	Description	Password Change Method
	K/M	Randomly generated by the system	Kerberos internal functional user. This user cannot be deleted, and its password cannot be changed. This internal account can only be used on nodes where Kerberos service is installed.	None
	kadmin/changepw	Randomly generated by the system		
	kadmin/history	Randomly generated by the system		
	krbtgt<*System domain name*>	Randomly generated by the system		
LDAP user	admin	None	FusionInsight Manager administrator. The primary group is **compcommon**, which does not have the group permission but has the permission of the **Manager_administrator** role.	The LDAP user cannot log in to the system, and the password cannot be changed.
	backup		The primary group is **compcommon**.	
	backup/manager		The primary group is **compcommon**.	
	oms		The primary group is **compcommon**.	
	oms/manager		The primary group is **compcommon**.	
	clientregister		The primary group is **compcommon**.	

User Type	Default User	Initial Password	Description	Password Change Method
	zookeeper		The primary group is **hadoop**.	
	zookeeper/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	zkcli		The primary group is **hadoop**.	
	zkcli/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	flume		The primary group is **hadoop**.	
	flume_server		The primary group is **hadoop**.	
	hdfs		The primary group is **hadoop**.	
	hdfs/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	mapred		The primary group is **hadoop**.	
	mapred/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	mr_zk		The primary group is **hadoop**.	
	mr_zk/hadoop.*<System domain name>*		The primary group is **hadoop**.	

User Type	Default User	Initial Password	Description	Password Change Method
	hue		The primary group is **supergroup**.	
	hive		The primary group is **hive**.	
	hive/hadoop.*<System domain name>*		The primary group is **hive**.	
	hive1		The primary group is **hive1**.	
	hive1/hadoop.*<System domain name>*		The primary group is **hive1**.	
	hive2		The primary group is **hive2**.	
	hive2/hadoop.*<System domain name>*		The primary group is **hive2**.	
	hive3		The primary group is **hive3**.	
	hive3/hadoop.*<System domain name>*		The primary group is **hive3**.	
	hive4		The primary group is **hive4**.	
	hive4/hadoop.*<System domain name>*		The primary group is **hive4**.	
	hbase		The primary group is **hadoop**.	

User Type	Default User	Initial Password	Description	Password Change Method
	hbase/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	thrift		The primary group is **hadoop**.	
	thrift/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	oozie		The primary group is **hadoop**.	
	hbase/zkclient.*<System domain name>*		The primary group is **hadoop**.	
	loader		The primary group is **hadoop**.	
	loader/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	spark2x		The primary group is **hadoop**.	
	spark2x/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	spark_zk		The primary group is **hadoop**.	
	spark2x1		The primary group is **hadoop**.	

User Type	Default User	Initial Password	Description	Password Change Method
	spark2x1/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	spark2x2		The primary group is **hadoop**.	
	spark2x2/hadoop.*<System domain name<*		The primary group is **hadoop**.	
	spark2x3		The primary group is **hadoop**.	
	spark2x3/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	spark2x4		The primary group is **hadoop**.	
	spark2x4/hadoop.*<System domain name>*		The primary group is **hadoop**.	
	kafka		The primary group is **kafkaadmin**.	
	kafka/hadoop.*<System domain name>*		The primary group is **kafkaadmin**.	
	storm		The primary group is **stormadmin**.	
	storm/hadoop.*<System domain name>*		The primary group is **stormadmin**.	

User Type	Default User	Initial Password	Description	Password Change Method
	storm_zk		The primary group is **storm**.	
	storm_zk/hadoop.*<System domain name>*		The primary group is **storm**.	
	kms/hadoop		The primary group is **kmsadmin**.	
	knox		The primary group is **compcommon**.	
	executor		The primary group is **compcommon**.	

NOTE

Log in to FusionInsight Manager, choose **System** > **Permission** > **Domain and Mutual Trust**, and check the value of **Local Domain**. In the preceding table, all letters in the system domain name contained in the username of the system internal user are lowercase letters.

For example, if **Local Domain** is set to **9427068F-6EFA-4833-B43E-60CB641E5B6C.COM**, the username of default HDFS startup user is **hdfs/hadoop.9427068f-6efa-4833-b43e-60cb641e5b6c.com**.

Database Users

The system database users include OMS database users and DBService database users.

Database Type	Default User	Initial Password	Description	Password Change Method
OMS database	ommdba	dbChangeMe@123456	OMS database administrator who performs maintenance operations, such as creating, starting, and stopping.	For details, see **Changing the Password for the OMS Database Administrator**.

Database Type	Default User	Initial Password	Description	Password Change Method
	omm	ChangeMe@123456	User for accessing OMS database data	For details, see **Changing the Password for the OMS Database Data Access User**.
DBService database	omm	dbserverAdmin@123	Administrator of the GaussDB database in the DBService component	For details, see **Changing the Password for a Component Database User**.
	hive	HiveUser@	User for Hive to connect to the DBService database **hivemeta**.	
	hive1	HiveUser@	User for Hive1 to connect to the DBService database **hivemeta1**.	
	hive2	HiveUser@	User for Hive2 to connect to the DBService database **hivemeta2**.	
	hive3	HiveUser@	User for Hive3 to connect to the DBService database **hivemeta3**.	
	hive4	HiveUser@	User for Hive4 to connect to the DBService database **hivemeta4**.	
	hive*NN*	HiveUser@	User for **Hive-***N* to connect to the DBService database **hive***N***meta** when multiple services are installed. For example, the user for **Hive-1** to connect to the DBService database **hive1meta** is **hive11**.	
	hue	HueUser@123	User for Hue to connect to the DBService database **hue**.	

Database Type	Default User	Initial Password	Description	Password Change Method
	sqoop	SqoopUser@	User for Loader to connect to the DBService database **sqoop**.	
	sqoop*N*	SqoopUser@	User for **Loader-***N* to connect to the DBService database **sqoop***N* when multiple services are installed. For example, the user for **Loader-1** to connect to the DBService database **sqoop1** is **sqoop1**.	
	oozie	OozieUser@	User for Oozie to connect to the DBService database **oozie**.	
	oozie*N*	OozieUser@	User for **Oozie-***N* to connect to the DBService database **oozie***N* when multiple services are installed. For example, the user for **Oozie-1** to connect to the DBService database **oozie1** is **oozie1**.	
	rangeradmin	Admin12!	User for Ranger to connect to the DBService database **ranger**.	

10.12.1.6 Default Permission Information

Role

Default Role	Description
Manager_administrator	Manager administrator who has all permissions for Manager. Manager administrators can create first-level tenants, create and modify user groups, and specify user permissions.

Default Role	Description
Manager_operator	Manager operator who has all the permissions on the **Homepage**, **Cluster**, **Hosts**, and **O&M** tab pages.
Manager_auditor	Manager auditor who has all permissions on the **Audit** tab page. Manager auditors can view and manage Manager system audit logs.
Manager_viewer	Manager viewer who has the permission to view information about **Homepage**, **Cluster**, **Hosts**, **Alarm**, **Event**, and **System > Permission**.
Manager_tenant	Manager tenant administrator. This role can create and manage sub-tenants for the non-leaf tenants to which the current user belongs. It has the permission to view alarms and events on **O&M > Alarm**.
System_administrator	System administrator, this role has Manager system administrator rights and all services administrator rights.
default	This role is the default role created for the **default** tenant. It has the management permissions on the Yarn component and the default queue. The default role of the default tenant that is not the first cluster to be installed is **c***<cluster ID>*_**default**.
Manager_administrator_180	FusionInsight Manager System administrator group. Internal system user group, which is used only between components.
Manager_auditor_181	FusionInsight Manager system auditor group. Internal system user group, which is used only between components.
Manager_operator_182	FusionInsight Manager system operator group. Internal system user group, which is used only between components.
Manager_viewer_183	FusionInsight Manager system viewer group. Internal system user group, which is used only between components.
System_administrator_186	System administrator group. Internal system user group, which is used only between components.
Manager_tenant_187	Tenant system user group. Internal system user group, which is used only between components.
default_1000	This group is created for tenant. Internal system user group, which is used only between components.

User group

Type	Default User Group	Description
OS User Group	hadoop	Users added to this group are granted the permission to submit all Yarn queue tasks.
	hadoopmanager	Users added to this user group can have the O&M manager rights of HDFS and Yarn. The O&M manager of HDFS can access the NameNode WebUI and perform active to standby switchover manually. The O&M manager of Yarn can access the ResourceManager WebUI, operate NodeManager nodes, refresh queues, and set node labels, but cannot submit tasks.
	hive	Common user group. Hive users must belong to this user group.
	hive1	Common user group. Hive1 users must belong to this user group.
	hive2	Common user group. Hive2 users must belong to this user group.
	hive3	Common user group. Hive3 users must belong to this user group.
	hive4	Common user group. Hive4 users must belong to this user group.
	kafka	Kafka common user group. A user in this group can access a topic only when a user in the kafkaadmin group grants the read and write permission of the topic to the user.
	kafkaadmin	Kafka administrator group. Users in this group have the rights to create, delete, authorize, read, and write all topics.
	kafkasuperuser	Topic read/write user group of Kafka. Users added to this group have the read and write permissions on all topics.
	storm	Users who are added to the storm user group can submit topologies and manage their own topologies.
	stormadmin	Users who are added to the stormadmin user group can have the storm administrator rights and can submit topologies and manage all topologies.
	supergroup	Users added to this user group can have the administrator rights of HBase, HDFS and Yarn and can use Hive.
	yarnviewgroup	Indicates the read-only user group of the Yarn task. Users in this user group can have the view permission on Yarn and MapReduce tasks.

Type	Default User Group	Description
	check_sec_ldap	Perform internal test on the active LDAP to see whether it works properly. This user group is generated randomly in a test and automatically deleted after the test is complete. Internal system user group, which is used only between components.
	compcommon	System internal group for accessing cluster system resources. All system users and system running users are added to this user group by default.
OS User Group	wheel	Primary group of the FusionInsight internal running user omm.
	ficommon	System common group that corresponds to **compcommon** for accessing cluster common resource files stored in the OS.

NOTE

If the current cluster is not the cluster that is installed for the first time in FusionInsight Manager, the default user group name of all components except Manager in the cluster is **c***<cluster ID>_ default user group name*, for example, **c2_hadoop**.

User

For details, see **User Account List**.

Service-related User Security Parameters

- **HDFS**

  The **dfs.permissions.superusergroup** parameter specifies the administrator group with the highest permission on the HDFS. The default value is **supergroup**.

- **Spark2x and Corresponding Multi-Instances**

  The **spark.admin.acls** parameter specifies the administrator list of the Spark2x. Members in the list are authorized to manage all Spark tasks. Users not added in the list cannot manage all Spark tasks. The default value is **admin**.

10.12.1.7 FusionInsight Manager Security Functions

You can query and set user rights data through the following FusionInsight Manager modules:

- User management: Users can be added, deleted, modified, queried, bound to user groups, and assigned with roles. For details, see **Managing Users**.
- User group management: User groups can be added, deleted, modified, queried, and bound to roles. For details, see **Managing User Groups**.

- Role management: Roles can be added, deleted, modified, queried, and assigned with the resource access rights of one or multiple components. For details, see **Managing Roles**.
- Tenant management: Tenants can be added, deleted, modified, queried, and bound to component resources. The system generates a role for each tenant to facilitate management. If a tenant is assigned with the rights of some resources, its corresponding role also has these rights. For details, see **Tenant Resources**.

10.12.2 Account Management

10.12.2.1 Account Security Settings

10.12.2.1.1 Unlocking LDAP Users and Management Accounts

Scenario

If the LDAP user cn=pg_search_dn,ou=Users,dc=hadoop,dc=com and LDAP management accounts cn=krbkdc,ou=Users,dc=hadoop,dc=com and cn=krbadmin,ou=Users,dc=hadoop,dc=com are locked, the administrator must unlock these accounts.

NOTE

If you input an incorrect password for the LDAP user or management account for five consecutive times, the LDAP user or management account is locked. The account is automatically unlocked after 5 minutes.

Procedure

Step 1 Log in to the active management node as user **omm** using the management IP address.

Step 2 Run the following command to switch the specified directory:

cd ${BIGDATA_HOME}/om-server/om/ldapserver/ldapserver/local/script

Step 3 Run the following command to unlock the LDAP user or management account:

./ldapserver_unlockUsers.sh *USER_NAME*

In the command, *USER_NAME* indicates the name of the user to be unlocked.

For example, to unlock the LDAP management **account cn=krbkdc,ou=Users,dc=hadoop,dc=com**, run the following command:

./ldapserver_unlockUsers.sh krbkdc

After the script is executed, enter the password of user **krbkdc** behind **ROOT_DN_PASSWORD**. If the following information is displayed, the account is successfully unlocked.

```
Unlock user krbkdc successfully.
```

----End

10.12.2.1.2 Unlocking an Internal System User

Scenario

If the service is abnormal, the internal user of the system may be locked. Please unlock the user promptly. Otherwise, the proper running of the cluster will be affected. For the list of system internal users, see **User Account List**. The internal user of the system cannot be unlocked using FusionInsight Manager.

Prerequisites

Obtain the default passwords of LDAP administrators **cn=root, dc=hadoop, and dc=com** based on the **User Account List** information list.

Procedure

Step 1 Use the following method to confirm whether the internal system username is locked:

1. oldap port number obtaining method:
   a. Log in to the FusionInsight Manager, select **System > OMS > oldap > Modify Configuration.**
   b. The **LDAP Listening Port** parameter value is **oldap port.**
2. Query domain name obtaining method:
   a. Log in to the FusionInsight Manager, select **System > Permission > Domain and Mutual Trust**.
   b. The **Local Domain** parameter value is the domain name.

      For example, the current system domain name is **9427068F-6EFA-4833-B43E-60CB641E5B6C.COM**.
3. Run the following command on each node in the cluster as user **omm** to query the number of password authentication failures:

   ldapsearch -H ldaps://*OMS_FLOAT_IP address:OLdap port* **-LLL -x -D cn=root,dc=hadoop,dc=com -b krbPrincipalName=***internal system username***@***domain name*,cn=*domain name*,**cn=krbcontainer,dc=hadoop,dc=com -w** *Password of LDAP administrator cn=root,dc=hadoop,dc=com* **-e ppolicy | grep krbLoginFailedCount**

   For example, query the number of password authentication failures for user **oms/manager**.

   ldapsearch -H ldaps://10.5.146.118:21750 -LLL -x -D cn=root,dc=hadoop,dc=com -b krbPrincipalName=oms/manager@9427068F-6EFA-4833-B43E-60CB641E5B6C.COM,cn=9427068F-6EFA-4833-B43E-60CB641E5B6C.COM,cn=krbcontainer,dc=hadoop,dc=com -w *cn=root,dc=hadoop,dc=com User Password* **-e ppolicy | grep krbLoginFailedCount**

   ```
   krbLoginFailedCount: 5
   ```
4. Log in to the FusionInsight Manager, select **System > Permission > Security Policy > Password Policy**.

5. View the Number of **Password Retries** parameter value, if the value is smaller than or equal to **krbLoginFailedCount,** the user is locked.

   NOTE

   You can also check whether internal users are locked by viewing operations logs.

Step 2 Log in to active management node as user omm, run the following command to unlock the user.

sh ${BIGDATA_HOME}/om-server/om/share/om/acs/config/unlockuser.sh --userName *internal system username*

For example,

sh ${BIGDATA_HOME}/om-server/om/share/om/acs/config/unlockuser.sh --userName oms/manager

----End

10.12.2.1.3 Enabling and Disabling Permission Verification on Cluster Components

Scenario

When the cluster is deployed in Security Mode or Normal Mode, HDFS and ZooKeeper verify the permission of users who attempt to access the services by default. Users without related permission cannot access resources in HDFS and ZooKeeper. When the cluster is deployed in Normal Mode, HBase and Yarn do not verify the permission of users who attempt to access the services by default. All users can access resources in HBase and Yarn.

Based on actual service requirements, the system administrator can enable permission verification on HBase and Yarn in the cluster in Normal Mode or disable permission verification on HDFS and ZooKeeper.

Impact on the System

After the permission verification is modified, the service configuration will expire. You need to restart the corresponding service for the configuration to take effect.

Procedure

Enable permission verification on HBase.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster >** *Name of the desired cluster* **> Services > HBase** > **Configurations**.

Step 3 Click **All Configurations**.

Step 4 Search for parameters **hbase.coprocessor.region.classes**, **hbase.coprocessor.master.classes**, and **hbase.coprocessor.regionserver.classes**.

Add the coprocessor parameter value **org.apache.hadoop.hbase.security.access.AccessController** to the end of the values of the preceding parameters, and separate the value from the original coprocessor parameter values by using a comma (,).

Step 5 Click **Save** and click **OK.**

When **Operation succeeded** is displayed, click **Finish**.

----End

Disable permission verification on HBase.

NOTE

After HBase permission verification is disabled, the existing permission data will be retained. If you want to delete permission information, disable permission verification, enter the HBase shell, and delete table **hbase:acl**.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations**.

Step 3 Click **All Configurations**.

Step 4 Search for parameters **hbase.coprocessor.region.classes**, **hbase.coprocessor.master.classes**, and **hbase.coprocessor.regionserver.classes**.

Delete the coprocessor parameter value **org.apache.hadoop.hbase.security.access.AccessController**.

Step 5 Click **Save** and click **OK**.

When **Operation succeeded** is displayed, click **Finish**.

----End

Disable permission verification on HDFS.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations**.

Step 3 Click **All Configurations**.

Step 4 Search for parameters **dfs.namenode.acls.enabled** and **dfs.permissions.enabled**.

- **dfs.namenode.acls.enabled** specifies whether the HDFS ACL is enabled. The default value is **true**, which indicates that the ACL is enabled. Change the value to **false**.
- **dfs.permissions.enabled** specifies whether the permission check is enabled on HDFS. The default value is **true**, which indicates that the permission check is enabled. Change the value to **false**. After the parameters are modified, the directories, owners and groups of files, and permission information in HDFS retain the same.

Step 5 Click **Save Configuration** and click **OK.**

When **Operation succeeded** is displayed, click **Finish**.

----End

Enable permission verification on Yarn.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster >** *Name of the desired cluster* **> Services > Yarn** > **Configurations**.

Step 3 Click **All Configurations**.

Step 4 Search for the parameter **yarn.acl.enable**.

yarn.acl.enable specifies whether the permission check is enabled on Yarn.

- In normal mode, the value is set to **false** by default to disable permission check. To enable permission check, change the value to **true**.
- In security mode, the value is set to **true** by default to enable authentication.

Step 5 Click **Save** and click **OK**.

When **Operation succeeded** is displayed, click **Finish**.

----End

Disable permission verification on ZooKeeper.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster >** *Name of the desired cluster* **> Services > ZooKeeper** > **Configurations**.

Step 3 Click **All Configurations**.

Step 4 Search for the parameter **skipACL**.

skipACL specifies whether the ZooKeeper permission check is skipped. The default value is **no**, which indicates that the permission check is used. Change the value to **yes**.

Step 5 Click **Save** and click **OK**.

When **Operation succeeded** is displayed, click **Finish**.

----End

10.12.2.1.4 Logging In to a Non-Cluster Node Using a Cluster User in Normal Mode

Scenario

When the cluster is installed in normal mode, the component clients do not support Kerberos authentication and cannot use the **kinit** command. Therefore, nodes outside the cluster cannot use users in the cluster by default. This may result in a user authentication failure when one of these nodes access a component server.

The node administrator can configure a user who has the same name as that of a user for a node outside the cluster, allow the user to log in to the node using the SSH protocol, and connect to the servers of components in the cluster by using the user who logs in to the OS.

Prerequisites

- The node outside the cluster can connect to the cluster service plane.

- The KrbServer service of the cluster is running properly.
- You have obtained the password of user **root** of the node outside the cluster.
- A **Human-machine** user has been planned and added to the cluster, and you have obtained the authentication credential file. For details, see **Creating a User** and **Exporting an Authentication Credential File**.

Procedure

Step 1 Log in to the node where a user is to be added as user **root**.

Step 2 Run the following commands:

rpm -qa | grep pam and **rpm -qa| grepkrb5-client**

The following RPM packages are displayed:

```
pam_krb5-32bit-2.3.1-47.12.1
pam-modules-32bit-11-1.22.1
yast2-pam-2.17.3-0.5.211
pam-32bit-1.1.5-0.10.17
pam_mount-32bit-0.47-13.16.1
pam-config-0.79-2.5.58
pam_krb5-2.3.1-47.12.1
pam-doc-1.1.5-0.10.17
pam-modules-11-1.22.1
pam_mount-0.47-13.16.1
pam_ldap-184-147.20
pam-1.1.5-0.10.17
krb5-client-1.6.3
```

Step 3 Check whether the RPM packages in the list are installed in the OS.

- If yes, go to **Step 5**.
- If no, go to **Step 4**.

Step 4 Obtain the lacked RPM packages from the OS image, upload the files to the current directory, and run the following command to install the RPM packages:

rpm -ivh *.rpm

NOTE

The RPM packages to be installed may bring security risks. The risks that may be brought by the installation of these RPM packages must be taken into consideration during OS hardening.

After the RPM packages are installed, go to **Step 5**.

Step 5 Run the following command to configure Kerberos authentication on PAM:

pam-config --add --krb5

NOTE

If you need to cancel Kerberos authentication and system user login on a non-cluster node, run the **pam-config --delete --krb5** command as user **root**.

Step 6 Decompress the authentication credential file to obtain **krb5.conf**, use WinSCP to upload this configuration file to the **/etc** directory on the node outside the cluster, and run the following command to configure related permission to enable other users to access the file, such as permission **604**:

chmod 604 /etc/krb5.conf

Step 7 Run the following command in the connection session as user **root** to add the corresponding OS user to the **Human-machine** user, and specify **root** as the primary group.

The OS user password is the same as the initial password when the **Human-machine** user is created on Manager.

useradd *Username* **-m-d /home/admin_test -g root -s /bin/bash**

For example, if the name of the **Human-machine** user is **admin_test**, run the following command:

useradd admin_test -m -d /home/admin_test -g root -s /bin/bash

NOTE

When you use the newly added OS user to log in to the node by using the SSH protocol for the first time, the system prompts that the password has expired after you enter the user password, and the system prompts that the password needs to be changed after you enter the user password again. You need to enter a new password that meets the password complexity requirements of both the node OS and the cluster.

----End

10.12.2.2 Changing the Password for a System User

10.12.2.2.1 Changing the Password for User admin

Scenario

The user **admin** is the system administrator account of FusionInsight Manager, periodically change the password for user **admin** to improve system security.

Procedure

Step 1 Log in to FusionInsight Manager.

Log in to the system as user **admin**.

Step 2 Move the cursor to **Hello, admin** in the upper right corner of the page.

In the displayed menu, click **Change Password**.

Step 3 Set **Old Password**, **New Password**, and **Confirm Password**, and click **OK**.

The password complexity requirements are as follows by default:

- The password ranges from 8 to 64 characters.
- The password must contain at least four types of the following: lowercase letters, uppercase letters, digits, spaces, and special characters which can only be ~`!?,.;-_'(){}[]/<>@#$%^&*+|\=.
- The password cannot be the same as the username or reverse username.
- The password cannot be a common password that is easy to crack.

- The password cannot be the same as the password that used in latest *N* times. *N* indicates the value of **Repetition Rule** in **Configuring Password Policies**.

----End

10.12.2.2.2 Changing the Password for an OS User

Scenario

During FusionInsight Manager installation, the system automatically creates user **omm** and **ommdba** on each node in the cluster. Periodically change the login passwords of the OS users **omm** and **ommdba** of the cluster node to improve the system O&M security.

The passwords of users **omm** and **ommdba** of the nodes can be different.

Prerequisites

- Obtain the IP address of the node where the passwords of users **omm** and **ommdba** are to be changed.
- You need to obtain the password of user **root** before modifying user **ommdba** and **omm**.

Change the password of an OS User

Step 1 Log in to the node where the password is to be changed as user **root**.

Step 2 Run the following command to change the user password:

passwd *ommdba*

Red Hat system displays the following information:

```
Changing password for user ommdba.
New password:
```

Step 3 Enter a new password. The policy for changing the password of an OS user varies according to the OS that is actually used.

```
Retype New Password:
Password changed.
```

----End

10.12.2.3 Changing the Password for a System Internal User

10.12.2.3.1 Changing the Password for the Kerberos Administrator

Scenario

Periodically change the password for the Kerberos administrator **kadmin** to improve the system O&M security.

If the user password is changed, the OMS Kerberos administrator password is changed as well.

Prerequisites

You have installed the client on any node in the cluster and obtain the IP address of the node.

Procedure

Step 1 Log in to the node where the client is installed as user root.

Step 2 Run the following command to go to the client directory, such as **/opt/hadoopclient**:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to change the password for kadmin/admin. The password changing takes effect on all servers.

kpasswd kadmin/admin

The password complexity requirements are as follows by default:

- The password contains at least 8 characters.
- The password must contain at least four types of the following: lowercase letters, uppercase letters, digits, spaces, and special characters which can only be ~`!?,.;-_'(){}[]/<>@#$%^&*+|\=.
- The password cannot be the same as the username or reverse username.
- The password cannot be a common password that is easy to crack, for example, **Admin@12345**.
- The password cannot be the same as the password that used in latest *N* times. *N* indicates the value of **Repetition Rule** in **Configuring Password Policies**.

----End

10.12.2.3.2 Changing the Password for the OMS Kerberos Administrator

Scenario

Periodically change the password for the OMS Kerberos administrator **kadmin** to improve the system O&M security.

If the user password is changed, the Kerberos administrator password is changed as well.

Procedure

Step 1 Log in to the management node using the management IP address as user **omm**.

Step 2 Run the following command to go to the related directory:

cd ${BIGDATA_HOME}/om-server/om/meta-0.0.1-SNAPSHOT/kerberos/scripts

Step 3 Run the following command to configure environment variables:

source component_env

Step 4 Run the following command to change the password for kadmin/admin. The password changing takes effect on all servers.

kpasswd kadmin/admin

The password complexity requirements are as follows by default:

- The password contains at least 8 characters.
- The password must contain at least four types of the following: lowercase letters, uppercase letters, digits, and special characters which can only be ~`!?,.;-_'(){}[]/<>@#$%^&*+|\=.
- The password cannot be the same as the username or reverse username.
- The password cannot be a common password that is easy to crack, for example, **Admin@12345**.
- The password cannot be the same as the password that used in latest *N* times. *N* indicates the value of **Repetition Rule** in **Configuring Password Policies**.

----End

10.12.2.3.3 Changing the Passwords of the LDAP Administrator and the LDAP User (Including OMS LDAP)

Scenario

It is recommended that the administrator periodically changes the passwords of LDAP administrator **cn=root,dc=hadoop,dc=com** and LDAP user **cn=pg_search_dn,ou=Users,dc=hadoop,dc=com** to improve the system O&M security.

If the passwords are changed, the password of the OMS LDAP administrator or user is changed as well.

NOTE

If the cluster is upgraded from an early version to a latest version, the LDAP administrator password will inherit the password policy of the old cluster. To ensure system security, you are advised to change the password after the cluster upgrade.

Impact on the System

- Changing the user password of the LdapServer service is a high-risk operation and requires restarting the KrbServer and LdapServer services. If KrbServer is restarted, users may fail to be queried by running the **id** command on nodes in the cluster temporarily. Therefore, exercise caution when restarting KrbServer.
- After the password of LDAP user **cn=pg_search_dn,ou=Users,dc=hadoop,dc=com** is changed, the user may be locked in the LDAP component. Therefore, you are advised to unlock the user after changing the password. For details about how to unlock the user, see **Unlocking LDAP Users and Management Accounts**.

Prerequisites

Before changing the password of LDAP user **cn=pg_search_dn,ou=Users,dc=hadoop,dc=com**, ensure that the user is not locked by running the following command on the active management node of the cluster:

NOTE

To query the OLDAP port number, perform the following steps:

1. Log in to FusionInsight Manager, choose **System** > **OMS** > **oldap** > **Modify Configuration**:
2. The value of **LDAP Service Listening Port** is the OLDAP port.

ldapsearch -H ldaps://*Floating IP address of OMS:OLDAP port***-LLL -x -D cn=pg_search_dn,ou=Users,dc=hadoop,dc=com -W -b cn=pg_search_dn,ou=Users,dc=hadoop,dc=com -e ppolicy**

Enter the password of the LDAP user **pg_search_dn**. If the following information is displayed, the user is locked. In this case, unlock the user. For details, see **Unlocking LDAP Users and Management Accounts**.

NOTE

The password of the LDAP user **pg_search_dn** is randomly generated by the system. You can obtain the password from the **/etc/sssd/sssd.conf or /etc/ldap.conf** file on the active node.

```
ldap_bind: Invalid credentials (49); Account locked
```

Procedure

Step 1 Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Service** > **LdapServer**.

Step 2 Choose **More** > **Change Database Password**. In the displayed dialog box, enter the password of the current login user and click **OK**.

Step 3 In the **Change Password** dialog box, select the user whose password to be modified in the **User Information** drop-down box.

Step 4 Enter the old password in the **Old Password** text box, and enter the new password in the **New Password** and **Confirm Password** text boxes.

The password must meet the following complexity requirements by default:

- Contains 16 to 32 characters.
- Contains at least three types of the following: uppercase letters, lowercase letters, digits, and special characters (`~!@#$%^&*()-_=+|[{}];,<.>/?).
- Cannot be the same as the username or the username spelled backwards.
- Cannot be the same as the current password.

Step 5 Select **I have read the information and understood the impact** and click **OK** to confirm the modification and restart the service.

----End

10.12.2.3.4 Changing the Password for the LDAP Administrator

Scenario

Periodically change the passwords of LDAP administrator accounts **cn=krbkdc**,**ou=Users**,**dc=hadoop**,**dc=com** and **cn=krbadmin**,**ou=Users**,**dc=hadoop**,**dc=com** to improve the system O&M security.If the user password is changed, the OMS LDAP administrator password is changed as well.

Impact on the System

1. You need to restart the KrbServer service after changing the password.
2. After the password is changed, check whether the LDAP management accounts cn=krbkdc,ou=Users,dc=hadoop,dc=com and cn=krbkdc,ou=Users,dc=hadoop,dc=com are locked. Run the following command on the active OMS node to check whether krbkdc is locked (similar method for krbadmin):

   NOTE

   oldap port number obtaining method:

   1. Log in to the FusionInsight Manager, select **System > OMS > oldap > Modify Configuration**.
   2. The **LDAP Listening Port** parameter value is oldap port.

   ldapsearch -H ldaps://*OMS_FLOAT_ IP address:OLdap port* **-LLL -x -D cn=krbkdc,ou=Users,dc=hadoop,dc=com -W -b cn= krbkdc,ou=Users,dc=hadoop,dc=com -e ppolicy**

   Enter the password for the LDAP management account **krbkdc**. If the following message is displayed, the account is locked. For details on how to unlock the account, see **Unlocking LDAP Users and Management Accounts**.

   ```
   ldap_bind: Invalid credentials (49); Account locked
   ```

Prerequisites

You have obtained the active management node IP address.

Procedure

Step 1 Log in to the management node using the active management IP address as user **omm**.

Step 2 Run the following command to go to the related directory:

cd ${BIGDATA_HOME}/om-server/om/meta-0.0.1-SNAPSHOT/kerberos/scripts

Step 3 Run the following command to change the password of the LDAP administrator accounts.

./okerberos_modpwd.sh

Enter the old password and enter a new password twice.

The password complexity requirements are as follows:

- The password ranges from 16 to 32 characters.
- The password must contain at least three types of the following: lowercase letters, uppercase letters, digits, and special characters which can only be `~!@#$%^&*()-_=+|[{}];,<.>/?.
- The password cannot be the same as the previous password.

If the following information is displayed, the password is changed successfully.

```
Modify kerberos server password successfully.
```

Step 4 Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **KrbServer** > **More** > **Restart Service**. Enter the password and do not select **Restart upper-layer services**. Click **OK** to restart the KrbServer service.

----End

10.12.2.3.5 Changing the Password for a Component Running User

Scenario

Periodically change the password for each component running user to improve the system O&M security.

Component running users can be classified into the following two types depending on whether their initial passwords are randomly generated by the system:

- If the initial password of a component running user is randomly generated by the system, the user is of the **Machine-Machine** type.
- If the initial password of a component running user is not randomly generated by the system, the user is of the **Human-Machine** type.

Impact on the System

All services need to be restarted for the password changing to take effect. The services are unavailable during the cluster restart.

Prerequisites

You have installed the client on any node in the cluster and obtain the IP address of the node.

Procedure

Step 1 Log in to the node where the client is installed as user **root**.

Step 2 Run the following command to go to the client directory, such as **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command and enter the password of user kadmin/admin to log in to the kadmin console:

kadmin -p kadmin/admin

NOTE

The default password of user **kadmin/admin** is **Admin@123,** which will expire upon your first login. Change the password as prompted and keep the new password secure.

Step 5 Run the following command to change the password of an internal system user. The password changing takes effect on all servers.

cpw *internal system username*

For example: **cpw oms/manager**

The password complexity requirements are as follows by default:

- The password contains at least 8 characters.
- The password must contain at least four types of the following: lowercase letters, uppercase letters, digits, spaces, and special characters which can only be ~`!?,.;-_'(){}[]/<>@#$%^&*+|\=.
- The password cannot be the same as the username or reverse username.
- The password cannot be a common password that is easy to crack, for example, **Admin@12345**.
- The password cannot be the same as the password that used in latest *N* times. *N* indicates the value of **Repetition Rule** in **Configuring Password Policies**. The policy affects only users of the **Human-Machine** type.

NOTE

Run the following command to check user information:

getprinc *internal system username*

For example: **getprinc oms/manager**

Step 6 Determine the type of the user whose password needs to be changed.

- If the user is a **Machine-Machine** user, perform **Step 7**.
- If the user is a **Human-Machine** user, the password is changed and no further action is required.

Step 7 Log in to FusionInsight Manager.

Step 8 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **More** > **Restart**.

Step 9 In the displayed window, enter the password of the current login administrator user and click **OK**.

Step 10 In the displayed dialog box, click **OK** to restart the cluster.

Step 11 After the system displays "**Operation succeeded**", click **Finish**. The cluster is successfully started.

----End

10.12.2.4 Changing the Password for a Database User

10.12.2.4.1 Changing the Password for the OMS Database Administrator

Scenario

Periodically change the password for the OMS database administrator to ensure the system O&M security.

Procedure

Step 1 Log in to the active management node using the active management IP address as user **root**.

NOTE

The password of user **ommdba** cannot be changed on the standby management node; otherwise, the cluster cannot work properly. The password of user **ommdba** can be changed after the system administrator performs operation only on the active management node. The operation does not need to be performed on the standby management node.

Step 2 Run the following command to switch to the user:

su - omm

Step 3 Run the following command to go to the directory:

cd $OMS_RUN_PATH/tools

Step 4 Run the following command to change the password for user **ommdba**:

mod_db_passwd ommdba

Step 5 Enter the old password of user **ommdba** and enter a new password twice. The password changing takes effect on all servers.

The password complexity requirements are as follows:

- The password ranges from 16 to 32 characters.
- The password must contain at least three types of the following: lowercase letters, uppercase letters, digits, and special characters which can only be ~`! @#$%^&*()-+_=\|[{}];",<.>/?
- The password cannot be the same as the username or reverse username.
- The password cannot be the same as the last 20 historical passwords.

If the following information is displayed, the password is changed successfully.

```
Congratulations, update [ommdba] password successfully.
```

----End

10.12.2.4.2 Changing the Password for the OMS Database Data Access User

Scenario

Periodically change the password for the OMS data access user to ensure the system O&M security.

Impact on the System

The OMS service needs to be restarted for the password changing to take effect. The cluster is unavailable during the restart.

Procedure

Step 1 Choose **System** > **OMS** > **gaussDB** > **Change Password** on FusionInsight Manager.

Step 2 Locate the row that contains user **omm** and click **Change Password** in the **Operation** to change the password for the OMS database user.

Step 3 In the displayed window, enter the password of the current login administrator user and click **OK**.

Step 4 Enter the old and new passwords as prompted.

The password complexity requirements are as follows:

- The password ranges from 8 to 32 characters.
- The password must contain at least three types of the following: lowercase letters, uppercase letters, digits, and special characters which can only be ~`!@#$%^&*()-+_=\|[{}];",<.>/?
- The password cannot be the same as the username or reverse username.
- The password cannot be the same as the last 20 historical passwords.

Step 5 Click **OK**. After the system displays **Operation succeeded**, click **Finish**.

Step 6 Locate the row that contains user **omm** and click **Restart OMS Service** in the **Operation** to restart the OMS database.

Step 7 In the displayed window, enter the password of the current login administrator user and click **OK**.

Step 8 In the dialog box that is displayed, click **OK**, and then restart the OMS service.

----End

10.12.2.4.3 Changing the Password for a Component Database User

Scenario

Periodically change the password for each component database user to improve the system O&M security.

Impact on the System

The services need to be restarted for the password changing to take effect. The services are unavailable during the restart.

Procedure

Step 1 Choose **Cluster** > *Name of the desired cluster* > **Services** on FusionInsight Manager.

Step 2 Determine the component database user whose password is to be changed.

To change the password for the other service database user, you must stop the service first, and go to **Step 3**.

Step 3 Click the service whose database user password is to be changed and choose **More** > **Change Database Password**. In the displayed window, enter the password of the current login administrator user and click **OK**.

Step 4 Enter the old and new passwords as prompted.

The password complexity requirements are as follows:

- The passwords of component databases contain 8 to 32 characters.
- The password must contain at least three types of the following: lowercase letters, uppercase letters, digits, and special characters which can only be ~`!@#$%^&*()-+_=\|[{}];",<.>/?
- The password cannot be the same as the username or reverse username.
- The password cannot be the same as the last 20 historical passwords.

Step 5 Select **I have read the information and understood the impact.** and click **OK**.

Step 6 After the password is changed, choose **More** > **Restart Service**. In the dialog box that is displayed, enter the password of the current login user, click **OK**, select **Restart upper-layer services**, and click **OK** to restart the service.

----End

10.12.2.4.4 Changing the Password for User omm in DBService

Step 1 Log in to the active DBService node as user **root**.

NOTE

The password of user **omm** for the DBService database cannot be changed on the standby DBService node. Change the password on the active DBService node only.

Step 2 Run the following command to switch the user.

su - omm

Step 3 Run the following command to switch the directory:

source $DBSERVER_HOME/.dbservice_profile

cd ${DBSERVICE_SOFTWARE_DIR}/sbin/

Step 4 Run the following command to change the password of user **omm**:

sh modifyDBPwd.sh

Step 5 Enter the old password of user **omm** and enter a new password twice.

The password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, and special characters (~`!@#$%^&*()-+_=\|[{}];",<.>/?).

- Must not be the same as the username or reverse username.
- The password cannot be the same as the last 20 historical passwords.

If the following information is displayed, the password is changed successfully:

```
Successful to modify password.
```

----End

10.12.3 Security Hardening

10.12.3.1 Hardening Policy

Hardening Tomcat

Tomcat is hardened as follows based on open-source software during FusionInsight Manager software installation and use:

- The Tomcat version is upgraded to the official version.
- Rights on directories under webapplications are set to 500. Some directories under webapplications support the write permission.
- The Tomcat installation package is automatically deleted after system software is installed.
- The automatic deployment function is disabled for projects under webapplications. Only three projects, web, cas and client-registry projects, are deployed.
- Some unused http methods are disabled, preventing attacks by using the http methods.
- The default shutdown port and command of the Tomcat server are changed to prevent hackers from shutting down the server and attacking servers and applications.
- To ensure security, the value of **maxHttpHeaderSize** is changed, which enables server administrators to control abnormal requests of clients.
- The Tomcat version description file is modified after Tomcat is installed.
- To prevent disclosure of Tomcat information, the Server attributes of Connector are modified so that attackers cannot obtain information about the server.
- Rights on files and directories of Tomcat, such as the configuration files, executable files, log directories, and temporary folders, are under control.
- Session facade recycling is disabled to prevent request leakage.
- LegacyCookieProcessor is used as CookieProcessor to prevent the leakage of sensitive data in cookies.

Hardening LDAP

LDAP is hardened as follows after a cluster is installed:

- In the LDAP configuration file, the password of the administrator account is encrypted using SHA. After the openldap is upgraded to 2.4.39 or later, data is

automatically synchronized between the active and standby LDAP nodes using the SASL External mechanism, which prevents disclosure of the password.

- The LDAP service in the cluster supports the SSLv3 protocol by default, which can be used safely. When the openldap is upgraded to 2.4.39 or later, the LDAP automatically users TLS1.0 or later to prevent unknown security risks.

Hardening JDK

- If the client process uses the AES256 encryption algorithm, JDK security hardening is required. The operations are as follows:

  Obtain the Java Cryptography Extension (JCE) package whose version matches that of JDK. The JCE package contains **local_policy.jar** and **US_export_policy.jar**. Copy the JAR files to the following directory and replace the files in the directory.

  Linux: **JDK installation directory/jre/lib/security**

  Windows: **JDK installation directory\jre\lib\security**

  NOTE

  Access the Open JDK open-source community to obtain the JCE file.

- If the client process uses the SM4 encryption algorithm, the JAR package needs to be updated.

  Obtain the **SMS4JA.jar** in **Client installation directory/JDK/jdk/jre/lib/ext/**, and copy the JAR package to the following directory:

  Linux: **JDK directory/jre/lib/ext/**

  Windows: **JDK directory\jre\lib\ext**

10.12.3.2 Configuring a Trusted IP Address to Access LDAP

Scenario

By default, the LDAP service deployed in the OMS and cluster can be accessed by any IP address. To enable the LDAP service to be accessed by only trusted IP addresses, you can configure the INPUT policy in the iptables filtering list.

Impact on the System

After the configuration, the LDAP service cannot be accessed by IP addresses that are not configured. Before the expansion, the added IP addresses need to be configured as trusted IP addresses.

Prerequisites

- You have collected the management plane IP addresses and service plane IP addresses of all nodes in the cluster and all floating IP addresses.
- You have obtained the root user account and password of all nodes in the cluster.

Procedure

Configure trusted IP addresses for the LDAP service on the OMS.

Step 1 Confirm the management node IP address. For details, see **Logging In to the Management Node**.

Step 2 Log in to FusionInsight Manager. For details, see **Logging In to FusionInsight Manager**.

Step 3 Choose **System** > **OMS**, and choose **oldap** > **Modify Configuration**, and view the OMS LDAP port number (which is the value of **LDAP Listening Port**). The default port number is **21750**.

Step 4 Log in to the active management node using the active management IP address as user **root**.

Step 5 Run the following command to view the INPUT policy in the iptables filtering list:

iptables -L

For example, when no rule is configured, the INPUT policy is displayed as follows:

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
```

Step 6 Run the following command to configure all IP addresses used by the cluster as trusted IP addresses. Each IP address needs to be added independently.

iptables -A INPUT -s *Trusted IP address* **-p tcp --dport** *Port number* **-j ACCEPT**

For example, to configure 10.0.0.1 as a trusted IP address and enable it to access port 21750, you need to run the following command:

iptables -A INPUT -s 10.0.0.1 -p tcp --dport 21750 -j ACCEPT

Step 7 Run the following command to configure all IP addresses as untrusted IP addresses. The trusted IP addresses will not be affected by this rule.

iptables -A INPUT -p tcp --dport *Port number* **-j DROP**

For example, to disable all IP addresses to access port 21750, you need to run the following command:

iptables -A INPUT -p tcp --dport 21750 -j DROP

Step 8 Run the following command to view the modified INPUT policy in the iptables filtering list:

iptables -L

For example, after a trusted IP address is configured, the INPUT policy is displayed as follows:

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  10.0.0.1             anywhere            tcp dpt:21750
DROP       tcp  --  anywhere             anywhere            tcp dpt:21750
```

Step 9 Run the following command to view the rules and rule numbers in the iptables filtering list:

iptables -L -n --line-number

```
Chain INPUT (policy ACCEPT)
num target     prot opt source               destination
1 DROP       tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:21750
```

Step 10 Run the following command to delete the desired rule from the iptables filtering list based on site requirement:

iptables -D INPUT *number of the rule to be deleted*

For example, to delete rule 1, run the following command:

iptables -D INPUT 1

Step 11 Log in to the standby management node using the standby management IP address as user **root**, and repeat **Step 5** to **Step 10**.

Configure trusted IP addresses for the LDAP service in the cluster.

Step 12 Log in to FusionInsight Manager.

Step 13 Choose **Cluster** > *Name of the desired cluster* > **Services** > **LdapServer** > **Instance**, and view the nodes where the LDAP services locate.

Step 14 Go to the **Configurations** page, and view the cluster LDAP port number (which is the value of **LDAP_SERVER_PORT**). The default port number is **21780**.

Step 15 Log in to the LDAP node using the LDAP service IP address as user **root**.

Step 16 Run the following command to view the INPUT policy in the iptables filtering list:

iptables -L

For example, when no rule is configured, the INPUT policy is displayed as follows:

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
```

Step 17 Run the following command to configure all IP addresses used by the cluster as trusted IP addresses. Each IP address needs to be added independently.

iptables -A INPUT -s *Trusted IP address* **-p tcp --dport** *Port number* **-j ACCEPT**

For example, to configure 10.0.0.1 as a trusted IP address and enable it to access port 21780, you need to run the following command:

iptables -A INPUT -s 10.0.0.1 -p tcp --dport 21780 -j ACCEPT

Step 18 Run the following command to configure all IP addresses as untrusted IP addresses. The trusted IP addresses will not be affected by this rule.

iptables -A INPUT -p tcp --dport *Port number* **-j DROP**

For example, to disable all IP addresses to access port 21780, you need to run the following command:

iptables -A INPUT -p tcp --dport 21780 -j DROP

Step 19 Run the following command to view the modified INPUT policy in the iptables filtering list:

iptables -L

For example, after a trusted IP address is configured, the INPUT policy is displayed as follows:

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
```

```
ACCEPT     tcp  --  10.0.0.1            anywhere            tcp dpt:21780
DROP       tcp  --  anywhere            anywhere            tcp dpt:21780
```

Step 20 Run the following command to view the rules and rule numbers in the iptables filtering list:

iptables -L -n --line-number

```
Chain INPUT (policy ACCEPT)
num target     prot opt source               destination
1 DROP       tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:21750
```

Step 21 Run the following command to delete the desired rule from the iptables filtering list based on site requirement:

iptables -D INPUT *number of the rule to be deleted*

For example, to delete rule 1, run the following command:

iptables -D INPUT 1

Step 22 Log in to the LDAP node using the IP address of another LDAP service IP address as user **root**, and repeat **Step 16** to **Step 21**.

----End

10.12.3.3 HFile and WAL Encryption

HFile and WAL Encryption

NOTICE

- Setting the HFile and WAL encryption mode to SMS4 or AES has a great impact on the system and will cause data loss in case of any misoperation. Therefore, this operation is not recommended.
- Batch data import using Bulkload does not support data encryption.

HFile and Write ahead log (WAL) in HBase are not encrypted by default. To encrypt them, perform the following operations.

Step 1 On any HBase node, run the following commands to create a key file as user **omm**:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh *<path>/hbase.jks <type> <length> <alias>*

- *<path>/hbase.jks* indicates the path of the generated jks file.
- **<type>** indicates the encryption type, which can be SMS 4 or AES.
- **<length>** indicates the key length. SMS 4 supports 16-bit and AES supports 128-bit.
- *<alias>* indicates the alias name of key file. When you create the key file for the first time, retain the default value **omm**.

For example, to generate an SMS4 encryption key, run the following command:

sh ${BIGDATA_HOME}/FusionInsight_HD_ 8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh /home/hbase/conf/hbase.jks SMS4 16 omm

To generate an AES encryption key, run the following command:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh /home/hbase/conf/hbase.jks AES 128 omm

NOTE

- The cluster operation user must have the **rw** permission of the *<path>/hbase.jks* directory. The directory requires already exists.
- After running the command, enter the same *<password>* four times. The password encrypted in **Step 3** is the same as the password in this step.

Step 2 Distribute the generated key files to the same directory on all nodes in the cluster and assign read and write permission to user **omm**.

NOTE

- Administrators need to select a safe procedure to distribute keys based on the enterprise security requirements.
- If the key files of some nodes are lost, repeat the step to copy the key files from other nodes.

Step 3 On FusionInsight Manager, set **hbase.crypto.keyprovider.parameters.encryptedtext** to the encrypted password. Set **hbase.crypto.keyprovider.parameters.uri** to the path and name of the key file.

- Format of **hbase.crypto.keyprovider.parameters.uri**: **jceks://** *<key_Path_Name>*.

  <key_Path_Name> indicates the path of the key file. For example, if the path of the key file is **/home/hbase/conf/hbase.jks**, set this parameter to **jceks:///home/hbase/conf/hbase.jks**.

- Format of **hbase.crypto.keyprovider.parameters.encryptedtext**: *<encrypted_password>*.

  <encrypted_password> indicates the encrypted password generated during the key file creation. The parameter value is displayed in ciphertext. Run the following command as user **omm** to obtain the related encrypted password on the nodes where HBase service is installed:

  sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh

  NOTE

  After running the command, enter the **<password>**. The password is the same as that in **Step 1**.

Step 4 On FusionInsight Manager, set **hbase.crypto.key.algorithm** to **SMS4** or **AES** to use SMS4 or AES for HFile encryption.

Step 5 On FusionInsight Manager, set **hbase.crypto.wal.algorithm** to **SMS4** or **AES** to use SMS4 or AES for WAL encryption.

Step 6 On FusionInsight Manager, set **hbase.regionserver.wal.encryption** to **true**.

Step 7 Save the settings and restart the HBase service for the settings to take effect.

Step 8 Set the encryption mode when creating an HBase table. **<type>** indicates the encryption type.

- When creating a table using commands, directly set the encryption mode to SMS4 or AES.

  create '*<table name>*', {*NAME => 'd'*, ***ENCRYPTION => '***<type>**'**}

- When creating a table using code, set the encryption mode to SMS4 or AES by adding the following information to the code:

```
public void testCreateTable() { String tableName = "user"; Configuration conf = getConfiguration();
HTableDescriptor htd = new HTableDescriptor(TableName.valueOf(tableName));

   HColumnDescriptor hcd = new HColumnDescriptor("info"); //Set the encryption mode to SMS4 or
AES. hcd.setEncryptionType("<type>"); htd.addFamily(hcd);

   HBaseAdmin admin = null; try { admin = new HBaseAdmin(conf);

      if(!admin.tableExists(tableName)) { admin.createTable(htd); } } catch (IOException e)
{ e.printStackTrace(); } finally { if(admin != null) { try { admin.close(); } catch (IOException e)
{ e.printStackTrace(); } } } }
```

Step 9 If you have configured SMS4 or AES encryption by performing **Step 1** to **Step 7** but do not set the related encryption parameter when creating the table in **Step 8**, the inserted data is not encrypted.

In this case, you can perform the following steps to encrypt inserted data:

1. Run the flush command for the table to import the data in the memory to the HFile.

   flush *'<table_name>'*

2. Run the following commands to modify the table properties:

   disable *'<table_name>'*

   alter *'<table_name>'*,**NAME=>** *'<column_name>'*,**ENCRYPTION =>** *'<type>'*

   enable *'<table_name>'*

3. Insert a new data record and flush the table.

   NOTE

   A new data record must be inserted so that the HFile will generate a new HFile and the unencrypted data inserted previously will be rewritten and encrypted.

   put *'<table_name>'*, **'id2','f1:c1','value22222222222222222222222222222222'**

   flush *'<table_name>'*

4. Perform the following step to rewrite the HFile:

   major_compact *'<table_name>'*

NOTICE

During this step, the HBase table is disabled and cannot provide services. Exercise caution when you perform this step.

----End

Modifying a Key File

NOTICE

Modifying a key file has a great impact on the system and will cause data loss in case of any misoperation. Therefore, this operation is not recommended.

During the **HFile and WAL Encryption** operation, the related key file must be generated and its password must be set to ensure system security. After a period of running, you can replace the key file with a new one to encrypt HFile and WAL.

Step 1 Run the following command to generate a new key file as user **omm**:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh *<path>/hbase.jks <type> <length> <alias-new>*

- *<path>/hbase.jks*: indicates the path of the generated **hbase.jks** file. The path and file name must be consistent with those of the key file generated in **HFile and WAL Encryption**.
- *<alias-new>*: indicates the alias of the key file. The alias must be different with that of the old key file.
- *<type>* indicates the encryption type, which can be SMS 4 or AES.
- *<length>* indicates the key length. SMS 4 supports 16-bit and AES supports 128-bit.

For example, to generate an SMS4 encryption key, run the following command:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh /home/hbase/conf/hbase.jks SMS4 16 omm_new

To generate an AES encryption key, run the following command:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh /home/hbase/conf/hbase.jks AES 128 omm_new

NOTE

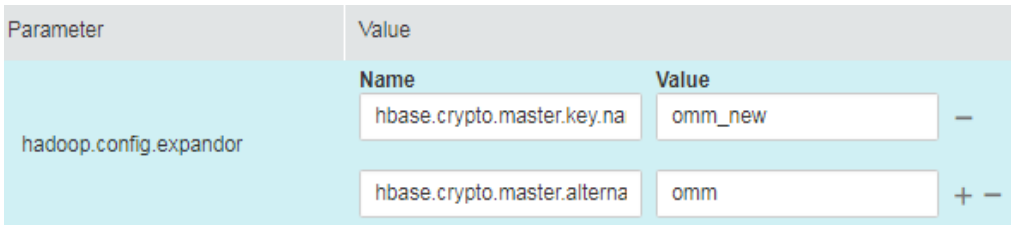
- The cluster operation user must have the **rw** permission of the *<path>/hbase.jks* directory. The directory requires already exists.
- After running the command, enter the same *<password>* three times. The password indicates the password of key files. The password of the old key file can be used, which does not cause any security risk.

Step 2 Distribute the generated key files to the same directory on all nodes in the cluster and assign read and write permission to user **omm**.

NOTE

Administrators need to select a safe procedure to distribute keys based on the enterprise security requirements.

Step 3 On the HBase service configuration page of FusionInsight Manager, add custom configuration items, set **hbase.crypto.master.key.name** to **omm_new**, set **hbase.crypto.master.alternate.key.name** to **omm**, and save the settings.

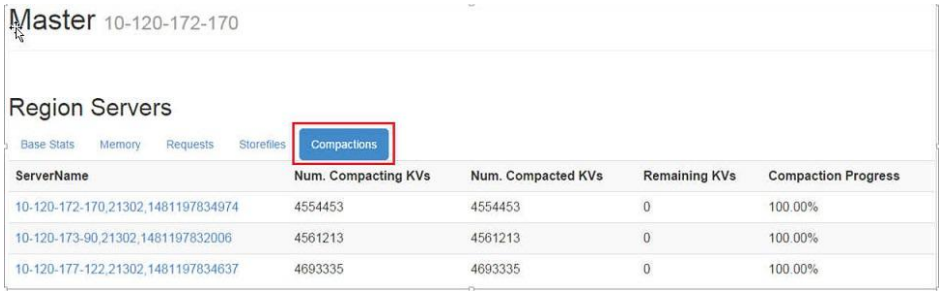


Step 4 Restart the HBase service for the configuration to take effect.

Step 5 In HBase shell, run the **major compact** command to generate the HFile file based on the new encryption algorithm.

major_compact *'<table_name>'*

Step 6 You can view the major compact progress from the HMaster web page.



Step 7 When all items in **Compaction Progress** reach **100%** and those in **Remaining KVs** are **0**, run the following command as user **omm** to destroy the old key file:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh *<path>/hbase.jks <alias-old>*

- *<path>/hbase.jks*: indicates the path of the generated **hbase.jks** file. The path and file name must be consistent with those of the key file generated.
- *<alias-old>*: indicates the alias of the old key file to be deleted.

For example:

sh ${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-HBase-2.2.3/hbase/bin/hbase-encrypt.sh /home/hbase/conf/hbase.jks omm

NOTE

The cluster operation user must have the **rw** permission for the *<path>/hbase.jks* directory. The directory requires already exists.

Step 8 Repeat **Step 2** and distribute the updated key files again.

Step 9 Delete the HBase self-defined configuration item **hbase.crypto.master.alternate.key.name** added in **Step 3** from FusionInsight Manager.

Step 10 Repeat **Step 4** for the configuration to take effect.

----End

10.12.3.4 Security Configuration

Configuring Security Channel Encryption

The channels between components are not encrypted by default. You can set the following parameters to configure security channel encryption.

Page access: On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > *component* > **Configurations**. Click **All Configurations**. Enter the parameter name in the search box.

NOTE

Restart the corresponding service after configuration.

Table 10-82 Parameter description

Parameter	Description	Default Value
hbase.rpc.protection	Indicates whether the HBase channels, including the remote procedure call (RPC) channels for HBase clients to access the HBase server and the RPC channels between the HMaster and RegionServer, are encrypted. If this parameter is set to **privacy**, the channels are encrypted and the authentication, integrity, and privacy functions are enabled. If this parameter is set to **integrity**, the channels are not encrypted and only the authentication and integrity functions are enabled. If this parameter is set to **authentication**, the channels are not encrypted, only packets are authenticated, and integrity and privacy are not required. **NOTE** The privacy mode encrypts transmitted content, including sensitive information such as user tokens, to ensure the security of the transmitted content. However, this mode has great impact on performance. Compared with the other two modes, this mode reduces read/write performance by about 60%. Modify the configuration based on the enterprise security requirements. The configuration items on the client and server must be the same.	-

Parameter	Description	Default Value
dfs.encrypt.data.transfer	Indicates whether the HDFS data transfer channels and the channels for clients to access HDFS are encrypted. The HDFS data transfer channels include the data transfer channels between DataNodes and the Data Transfer (DT) channels for clients to access DataNodes. The value **true** indicates that the channels are encrypted. The channels are not encrypted by default.	**false**
dfs.encrypt.data.transfer.algorithm	Indicates whether the HDFS data transfer channels and the channels for clients to access HDFS are encrypted. This parameter is valid only when dfs.encrypt.data.transfer is set to true. The default value is **3des**, which indicates that the 3DES algorithm is used for encryption. The value can also be set to **rc4**; however, to avoid security risks, do not set the parameter to this value.	**3des**
hadoop.rpc.protection	Indicates whether the RPC channels of each module in Hadoop are encrypted. The channels include: ● RPC channels for clients to access HDFS ● RPC channels between modules in HDFS, for example, RPC channels between DataNode and NameNode ● RPC channels for clients to access YARN ● RPC channels between NodeManager and ResourceManager ● RPC channels for Spark to access YARN and HDFS ● RPC channels for MapReduce to access YARN and HDFS ● RPC channels for HBase to access HDFS The **privacy** indicates that the channels are encrypted by default. The **authentication** indicates that channels are not encrypted. **NOTE** You can set this parameter on the HDFS component configuration page. The parameter setting is valid globally, that is, the setting of whether the RPC channel is encrypted takes effect on all modules in Hadoop.	● Security mode: **privacy** ● Normal Mode: **authentication**

Setting the Maximum Number of Concurrent Web Connections

To ensure Web server security, the number of new connections is limited when the number of user connections reaches a specific threshold. This prevents DDOS attacks and service unavailability caused by too many users accessing the web server at the same time.

Page access:

Page access: On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > *component* > **Configurations**. Click **All Configurations**. Enter the parameter name in the search box.

Table 10-83 Parameter description

Parameter	Description	Default Value
hadoop.http.server.MaxRequests	Specifies the maximum number of concurrent web connections of each component. Components include HDFS and YARN.	2000
spark.connection.maxRequest	Maximum number of request connections of JobHistory.	5000

10.12.3.5 Configuring an IP Address Whitelist for Modifications Allowed by HBase

If the Replication function is enabled for HBase clusters, a protection mechanism for data modification is added on the standby HBase cluster to ensure data consistency between the active and standby clusters. Upon receiving an RPC request for data modification, the standby HBase cluster checks the permission of the user who sends the request (only HBase manage users have the modification permission). Then it checks the validity of the source IP address of the request. Only modification requests from IP addresses in the white list are accepted. The IP address white list is configured by the **hbase.replication.allowedIPs** item.

Page access: On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > *component* > **Configurations**. Click **All Configurations**. Enter the parameter name in the search box.

Table 10-84 Parameter description

Parameter	Description	Default Value
hbase.replication.allowedIPs	Allows replication request processing from configured IP addresses only. It supports comma separated regex patterns. Each pattern can be any of the following: • Regex Pattern eg: 10.18.40.* , 10.18.* , 10.18.40.11 • Range Pattern (Range can be specified only in the last octet) eg: 10.18.40.[10 to 20] If this item is empty (default value), the white list contains only the IP address of the RegionServer of the cluster, indicating that only modification requests from the RegionSever of the standby HBase cluster are accepted.	N/A

10.12.3.6 Updating a Key for a Cluster

Scenario

When a cluster is installed, an encryption key is generated automatically by the system so that the security information in the cluster (such as all database user passwords and key file access passwords) can be stored in encryption mode. After the cluster is installed, if the original key is accidentally disclosed or a new key is required, you can perform the following operations to manually update the key.

Impact on the System

- After a cluster key is updated, a new key is generated randomly in the cluster. This key is used to encrypt and decrypt the newly stored data. The old key is not deleted, and it is used to decrypt old encrypted data. After security information is modified, for example, a database user password is changed, the new password is encrypted using the new key.
- When a key is updated for a cluster, the cluster must be stopped and cannot be accessed.

Prerequisites

- You have obtained the IP addresses of the active and standby management nodes. For details, see **Logging In to the Management Node**.
- You have stopped the upper-layer service applications that depend on the cluster.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Stop**, and enter the password of the current login administrator for authentication.

In the displayed window, click **OK**. **Operation succeeded** is displayed. Click **Finish**. The cluster is stopped.

Step 3 Log in to the active management node as user **omm** with the IP address of the active management node.

Step 4 Run the following command to prevent you from being forcibly logged out when a timeout occurs:

TMOUT=0

NOTE

After the operations in this section are complete, run the **TMOUT=***Timeout interval* command to restore the timeout interval in a timely manner. For example, **TMOUT=600** indicates that a user is logged out if the user does not perform any operation within 600 seconds.

Step 5 Run the following command to switch the directory:

cd ${BIGDATA_HOME}/om-server/om/tools

Step 6 Run the following command to update the cluster key:

sh updateRootKey.sh

Enter **y** as prompted.

```
The root key update is a critical operation.
Do you want to continue?(y/n):
```

If the following information is displayed, the key is updated successfully.

```
Step 4-1: The key save path is obtained successfully.
...
Step 4-4: The root key is sent successfully.
```

Step 7 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Start**.

In the confirmation dialog box, click **OK** to start the cluster. **Operation succeeded** is displayed. Click **Finish**. The cluster is started.

----End

10.12.3.7 Hardening the LDAP

Configure the LDAP firewall policy.

In the cluster adopting the dual-plane networking, the LDAP is deployed on the service plane. To ensure the LDAP data security, you are advised to configure the firewall policy for the whole cluster to disable relevant LDAP ports.

Step 1 Log in to FusionInsight Manager.

Step 2 Click **Cluster** > *Name of the desired cluster* > **Services** > **LdapServer** > **Configurations**.

Step 3 Check the value of **LDAP_SERVER_PORT**, which is the service port of LdapServer.

Step 4 To ensure data security, configure the firewall policy for the whole cluster to disable the LdapServer port based on the customer's firewall environment.

----End

Enable the LDAP Audit Log Output.

Users can set the audit log output level of the LDAP service and output audit logs in a specified directory, for example, **/var/log/messages**. The logs output can be used to check user activities and operation commands.

NOTE

If the function of LDAP audit log output is enabled, massive logs are generated, affecting the cluster performance. Exercise caution when enabling this function.

Step 1 Log in to any LdapServer node.

Step 2 Run the following command to edit the **slapd.conf.consumer** file, and set the value of **loglevel** to **256** (You can view the log level definition by running the **man slapd.conf** command on the OS).

cd ${BIGDATA_HOME}/FusionInsight_BASE_8.1.0.1/install/FusionInsight-ldapserver-2.7.0/ldapserver/local/template

vi slapd.conf.consumer

```
...
pidfile         [PID_FILE_SLAPD_PID]
argsfile        [PID_FILE_SLAPD_ARGS]
loglevel 256
...
```

Step 3 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **LdapServer** > **More** > **Restart Service**, enter the administrator password, and restart the service.

----End

10.12.3.8 Configuring Kafka Data Encryption During Transmission

Scenario

Data between the Kafka client and the broker is transmitted in plain text. The Kafka client may be deployed in an untrusted network, exposing the transmitting data to leakage and tampering risks.

Procedure

The channel between components is not encrypted by default. You can set the following parameters to enable security channel encryption.

Navigation path for setting parameters: On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Kafka** > **Configuration**. On the

displayed page, click the **All Configurations** tab. Enter a parameter name in the search box.

NOTE

After the configuration, restart the corresponding service for the settings to take effect.

Table 10-85 describes the parameters related to transmission encryption on the Kafka server.

Table 10-85 Parameters relevant to Kafka data encryption during transmission

Parameter	Description	Default Value
ssl.mode.enable	Indicates whether to enable the Secure Sockets Layer (SSL) protocol. If this parameter is set to **true**, services relevant to the SSL protocol are started during the broker startup.	false
security.inter.broker.protocol	Indicates communication protocol between brokers. The communication protocol can be PLAINTEXT, SSL, SASL_PLAINTEXT, or SASL_SSL.	SASL_PLAINTEXT

The SSL protocol can be configured for the server or client to encrypt transmission and communication only after **ssl.mode.enable** is set to **true** and broker enables the **SSL** and **SASL_SSL** protocols.

10.12.3.9 Configuring HDFS Data Encryption During Transmission

Configuring HDFS Security Channel Encryption

The channel between components is not encrypted by default. You can set parameters to enable security channel encryption.

Navigation path for setting parameters: On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations**. On the displayed page, click the **All Configurations** tab. Enter a parameter name in the search box.

NOTE

After the configuration, restart the corresponding service for the settings to take effect.

Table 10-86 Parameters

Configuration Item	Description	Default Value
hadoop.rpc.protection	**NOTICE** • The setting takes effect only after the service is restarted. Rolling restart is not supported. • After the setting, you need to download the client configuration file again. Otherwise, HDFS cannot provide the read and write services. Indicates whether the RPC channels of each module in Hadoop are encrypted. The channels include: • RPC channels for clients to access HDFS • RPC channels between modules in HDFS, for example, between DataNode and NameNode • RPC channels for clients to access Yarn • RPC channels between NodeManager and ResourceManager • RPC channels for Spark to access Yarn and HDFS • RPC channels for MapReduce to access Yarn and HDFS • RPC channels for HBase to access HDFS **NOTE** The setting takes effect globally, that is, the encryption attribute of the RPC channel of each module in the Hadoop takes effect.	• Security mode: privacy • Normal mode: authentication **NOTE** • **authentication**: indicates that only authentication is required. • **integrity**: indicates that authentication and consistency check need to be performed. • **privacy**: indicates that authentication, consistency check, and encryption need to be performed.

Configuration Item	Description	Default Value
dfs.encrypt.data.transfer	Indicates whether the HDFS data transfer channels and the channels for clients to access HDFS are encrypted. The HDFS data transfer channels include the data transfer channels between DataNodes and the Data Transfer (DT) channels for clients to access DataNodes. The value **true** indicates that the channels are encrypted. The channels are not encrypted by default. **NOTE** • This parameter is valid only when **hadoop.rpc.protection** is set to **privacy**. • If a large amount of service data is transmitted, enabling encryption by default severely affects system performance. • If data transmission encryption is configured for one cluster in the trusted cluster, the same data transmission encryption must be configured for the peer cluster.	false
dfs.encrypt.data.transfer.algorithm	Indicates the algorithm to encrypt the HDFS data transfer channels and the channels for clients to access HDFS. This parameter is valid only when **dfs.encrypt.data.transfer** is set to **true**. **NOTE** The default value is **3des**, indicating that 3DES algorithm is used to encrypt data. The value can also be set to **rc4**. However, to avoid security risks, you are not advised to set the parameter to this value.	3des
dfs.encrypt.data.transfer.cipher.suites	This parameter can be left empty or set to **AES/CTR/NoPadding** to specify the cipher suite for data encryption. If this parameter is not specified, the encryption algorithm specified by **dfs.encrypt.data.transfer.algorithm** is used for data encryption. The default value is **AES/CTR/NoPadding**.	AES/CTR/NoPadding

10.12.3.10 Encrypting the Communication Between Controller and Agent

Scenario

After a cluster is installed, Controller and Agent need to communicate with each other. The Kerberos authentication is used during the communication. By default, the communication is not encrypted during the communication for the sake of cluster performance. Users who have demanding security requirements can use the method described in this section for encryption.

Impact on the System

- Controller and all Agents automatically restart, which interrupts FusionInsight Manager.
- The performance of management nodes decreases in large clusters. You are advised to enable the encryption function for clusters with a maximum of 200 nodes.

Prerequisites

You have obtained the IP addresses of the active and standby management nodes.

Procedure

Step 1 Log in to the active management node as user **omm**.

Step 2 Run the following command to disable user logout on system timeout:

TMOUT=0

NOTE

After the operations in this section are complete, run the **TMOUT=***Timeout interval* command to restore the timeout interval in a timely manner. For example, **TMOUT=600** indicates that a user is logged out if the user does not perform any operation within 600 seconds.

Step 3 Run the following command to switch the directory:

cd ${CONTROLLER_HOME}/sbin

Step 4 Run the following command to enable communication encryption:

./enableRPCEncrypt.sh -t

Run the **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** command to check whether **ResHAStatus** of the active management node Controller is **Normal** and whether you can log in to FusionInsight Manager again. If yes, the enablement is successful.

Step 5 Run the following command to disable communication encryption when necessary:

./enableRPCEncrypt.sh -f

Run the **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** command to check whether **ResHAStatus** of the active management node Controller is **Normal**

and whether you can log in to FusionInsight Manager again. If yes, the disablement is successful.

----End

10.12.3.11 Updating SSH Keys for User omm

Scenario

During cluster installation, the system automatically generates the SSH public key and private key for user **omm** to establish the trust relationship between nodes. After the cluster is installed, if the original keys are accidentally disclosed or new keys are used, the system administrator can perform the following operations to manually change the keys.

Prerequisites

- The cluster has been stopped.
- No other management operations are being performed.

Procedure

Step 1 Log in as user omm to the node whose SSH keys need to be replaced.

If the node is a Manager management node, run the following command on the active management node.

Step 2 Run the following command to disable user logout upon system timeout:

TMOUT=0

NOTE

After the operations in this section are complete, run the **TMOUT=***Timeout interval* command to restore the timeout interval in a timely manner. For example, **TMOUT=600** indicates that a user is logged out if the user does not perform any operation within 600 seconds.

Step 3 Run the following command to generate a key for the node:

- If the node is a Manager management node, run the following command:

  sh ${CONTROLLER_HOME}/sbin/update-ssh-key.sh

- If the node is a non-Manager management node, run the following command:

  sh ${NODE_AGENT_HOME}/bin/update-ssh-key.sh

If **Succeed to update ssh private key.** is displayed when the preceding command is executed, the SSH key is generated successfully.

Step 4 Run the following command to copy the public key of the node to the active management node:

scp ${HOME}/.ssh/id_rsa.pub** *oms_ip*:**${HOME}/.ssh/id_rsa.pub_bak

oms_ip: indicates the IP address of the active management node.

Enter the password of user **omm** to copy the files.

Step 5 Log in to the active management node as user **omm**.

Step 6 Run the following command to disable user logout on system timeout:

TMOUT=0

Step 7 Run the following command to switch the directory:

cd ${HOME}/.ssh

Step 8 Run the following command to add new public keys:

cat id_rsa.pub_bak >> authorized_keys

Step 9 Run the following command to move the temporary public key file, for example, **/tmp**.

mv -f id_rsa.pub_bak /tmp

Step 10 Copy the **authorized_keys** file of the active management node to the other nodes in the cluster:

scp authorized_keys *node_ip*:**/${HOME}/.ssh/authorized_keys**

node_ip: indicates the IP address of another node in the cluster. Multiple IP addresses are not supported.

Step 11 Run the following command to confirm private key replacement without entering the password:

ssh *node_ip*

node_ip: indicates the IP address of another node in the cluster. Multiple IP addresses are not supported.

Step 12 Log in to the FusionInsight Manager. On the **Homepage** page, click > **Start** to start the cluster.

----End

10.12.4 Security Maintenance

10.12.4.1 Account Maintenance Suggestions

You are advised to perform routine checks on accounts. The check covers the following items:

- Check whether the accounts of the OS, FusionInsight Manager, and each component are necessary and whether temporary accounts have been deleted.
- Check whether the permissions of the accounts are appropriate. Different administrators have different rights.
- Check and audit the logins and operation records of all types of accounts.

10.12.4.2 Password Maintenance Suggestions

User identity authentication is a must for accessing the application system. The complexity and validity period of user accounts and passwords must meet customers' security requirements.

The password maintenance suggestions are as follows:

1. Dedicated personnel must be arranged to manage the OS password.
2. The passwords must meet the complexity requirements, such as minimum password length or character types.
3. Passwords must be encrypted before transfer. Generally, do not transfer passwords using emails.
4. Passwords must be encrypted in configuration files.
5. Enterprise users need to change the passwords when the system is handed over.
6. Passwords must be periodically changed.

10.12.4.3 Logs Maintenance Suggestions

Operation logs help discover exceptions such as illegal operations and login by unauthorized users. The system records important operations in logs. You can use operation logs to locate problems.

Checking Logs Regularly

Check system operation logs periodically and handle exceptions such as unauthorized operations or logins in a timely manner.

Backing Up Logs Regularly

The audit logs provided by FusionInsight Manager and cluster record the user activities and operations. You can export the audit logs on FusionInsight Manager. If there are too many audit logs in the system, you can configure dump parameters to dump audit logs to a specified server to ensure that the cluster nodes disk space is sufficient.

Maintenance Owner

Network monitoring engineers and system maintenance engineers

10.12.5 Security Statement

JDK Usage Statement

MRS is a big data cluster that provides distributed data analysis and computing capabilities for users. OpenJDK is the built-in JDK of MRS, which is mainly applied in the following scenarios:

- Performing O&M for platform services
- Running the Linux client (mostly for service request submission and application O&M)

JDK Risk Statement

The system implements permission control on the built-in JDK. Only users in related groups of the FusionInsight platform can access the JDK. In addition, the platform is deployed on a customer's intranet, which has low security risks.

JDK Hardening

For details about how to harden the JDK, see "Hardening JDK" in **Hardening Policy**.

Public IP Addresses in Hue

Hue uses the test cases of third-party packages, such as **ipadrress**, **requests**, and **Django**, and uses the public IP addresses in the comments of the test cases. However, these public IP addresses are not involved when Hue provides services, and the Hue configuration file does not involve these public IP addresses.

10.13 Alarm Reference (Applicable to MRS 3.*x*)

10.13.1 ALM-12001 Audit Log Dumping Failure

Description

Cluster audit logs need to be dumped on a third-party server due to the local historical data backup policy. The system starts to check the dump server at 3 a.m. every day. If the dump server meets the configuration conditions, audit logs can be successfully dumped. This alarm is generated when the audit log dump fails if the disk space of the dump directory on the third-party server is insufficient or a user changes the username, password, or dump directory of the dump server.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12001	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.

Impact on the System

System can store a maximum of only 50 dump files locally. If the fault persists on the dump server, the local audit logs may be lost.

Possible Causes

- The network connection is abnormal.
- The username, password, or dump directory of the dump server does not meet the configuration conditions.
- The disk space of the dump directory is insufficient.

Procedure

Check whether the network connection is normal.

Step 1 On the FusionInsight Manager home page, choose **Audit > Configurations**.

Step 2 Check whether the SFTP IP on the dump configuration page is valid.

Log in to the node where Manager is located as user **root** and run the **ping** command to check whether the network connection between the SFTP server and the cluster is normal.

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 Repair the network connection, reset the SFTP password, and click **OK**.

Step 4 Wait for 2 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the username, password, or dump directory are correct.

Step 5 On the dump configuration page, check whether the username, password, and dump directory of the third-party server are correct.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 Change the username, password, or dump directory, reset the SFTP password and click **OK**.

Step 7 Wait for 2 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check whether the disk space of the dump directory is sufficient.

Step 8 Log in to the third-party server as user **root** and run the **df** command to check whether the disk space of the dump directory of the third-party server exceeds 100 MB.

- If yes, go to **Step 11**.
- If no, go to **Step 9**.

Step 9 Expand disk space capacity for the third-party server, Reset the SFTP password and click **OK**

Step 10 Wait for 2 minutes, view real-time alarms and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Reset the dump rule.

Step 11 On the FusionInsight Manager home page, choose **Audit > Configurations**.

Step 12 Reset dump rules, set the parameters properly, and click **OK**.

Step 13 Wait for 2 minutes, view real-time alarms and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Collect fault information.

Step 14 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 15 Select **OmmServer** from the **Service** and click **OK**.

Step 16 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 17 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.2 ALM-12004 OLdap Resource Abnormal

Description

The system checks LDAP resources every 60 seconds. This alarm is generated when the system detects that the LDAP resources in Manager are abnormal for six consecutive times.

This alarm is cleared when the Ldap resource in the Manager recovers and the alarm handling is complete.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12004	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Manager and component WebUI authentication services are unavailable and cannot provide security authentication and user management functions for web upper-layer services. Users may be unable to log in to the WebUIs of Manager and components.

Possible Causes

The LdapServer process in the Manager is abnormal.

Procedure

Check whether the LdapServer process in the Manager is normal.

Step 1 Log in the Manager node in the cluster as user **omm**.

Log in to FusionInsight Manager using the floating IP address, and run the **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** command to check the information about the current Manager two-node cluster.

Step 2 Run **ps -ef | grep slapd** command to check whether the LdapServer resource process in the **${BIGDATA_HOME}/om-server/om/** in the process configuration file is running properly.

NOTE

You can determine that the resource is normal by checking the following information:

1. After the **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** command runs, **ResHAStatus** of the OLdap is **Normal**.
2. After the **ps -ef | grep slapd** command runs, the slapd process of port 21750 can be viewed.
   - If yes, go to **Step 3**.
   - If no, go to **Step 4**.

Step 3 Run the **kill -2** *ldap pid* command to restart the LdapServer process and wait for 20 seconds. The HA starts the OLdap process automatically. Check whether the current OLdap resource is in normal state.

- If yes, the operation is complete.
- If no, go to **Step 4**.

Collect fault information.

Step 4 On the FusionInsight Manager home page, choose **O&M** > **Log > Download**.

Step 5 Select **OmsLdapServer** and **OmmServer** from the **Service** and click **OK**.

Step 6 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.3 ALM-12005 OKerberos Resource Abnormal

Description

The alarm module checks the status of the Kerberos resource in Manager every 80 seconds. This alarm is generated when the alarm module detects that the Kerberos resources are abnormal for six consecutive times.

This alarm is cleared when the Kerberos resource recovers and the alarm handling is complete.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The component WebUI authentication services are unavailable and cannot provide security authentication functions for web upper-layer services. Users may be unable to log in to FusionInsight Manager and the WebUIs of components.

Possible Causes

The OLdap resource on which the Okerberos depends is abnormal.

Procedure

Check whether the OLdap resource on which the Okerberos depends is abnormal in the Manager.

Step 1 Log in the Manager node in the cluster as user **omm**.

Log in to FusionInsight Manager using the floating IP address, and run the **sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh** command to check the information about the current Manager two-node cluster.

Step 2 Run the **sh ${BIGDATA_HOME}/om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the OLdap resource status managed by HA is normal. (In single-node mode, the OLdap resource is in the Active_normal state; in the two-node mode, the OLdap resource is in the Active_normal state on the active node and in the Standby_normal state on the standby node.)

- If yes, go to **Step 4**.
- If no, go to **Step 3**.

Step 3 See the procedure in **ALM-12004 OLdap Resource Abnormal** to resolve the problem. After the OLdap resource status recovers, check whether the OKerberos resource status is normal.

- If yes, the operation is complete.

- If no, go to **Step 4**.

Collect fault information.

Step 4 On the FusionInsight Manager home page, choose **O&M** > **Log** > **Download**.

Step 5 Select **OmsKerberos** and **OmmServer** from the **Service** and click **OK**.

Step 6 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.4 ALM-12006 Node Fault

Description

Controller checks NodeAgent heartbeat messages every 30 seconds. If Controller does not receive heartbeat messages from a NodeAgent, it attempts to restart the NodeAgent process. This alarm is generated if the NodeAgent fails to be restarted for three consecutive times.

This alarm is cleared when Controller can properly receive the status report of the NodeAgent.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Services on the node are unavailable.

Possible Causes

The network is disconnected, the hardware is faulty, or the operating system runs slowly.

Procedure

Check whether the network is disconnected, whether the hardware is faulty, or whether the operating system runs slowly.

Step 1 In the FusionInsight Manager portal, click **O&M > Alarm > Alarms**, click ⌄ in the row where the alarm is located , and click the host name to view the host address for which the alarm is generated.

Step 2 Log in to the active management node as user **root**.

Step 3 Run the **ping** *IP address of the faulty host* command to check whether the faulty node is reachable.

- If yes, go to **Step 12**.
- If no, go to **Step 4**.

Step 4 Contact the network administrator to check whether the network is faulty.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Recover the network fault and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Step 6 Contact the system administrator to check whether the node hardware (CPU or memory) is faulty.

- If yes, go to **Step 7**.
- If no, go to **Step 12**.

Step 7 Repair or replace the faulty components and restart the node. Then check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Step 8 If a large number of node faults are reported in the cluster, the floating IP address resource may be abnormal. As a result, the controller cannot detect the agent heartbeat.

Log in to any management node and view the **/var/log/Bigdata/omm/oms/ha/scriptlog/floatip.log** file to check whether the logs generated one to two minutes before and after the fault occurs are complete.

For example, a complete log is in the following format:

```
2017-12-09 04:10:51,000 INFO (floatip) Read from ${BIGDATA_HOME}/om-server_8.1.0.1/om/etc/om/
routeSetConf.ini,value is : yes
2017-12-09 04:10:51,000 INFO (floatip) check wsNetExport : eth0 is up.
2017-12-09 04:10:51,000 INFO (floatip) check omNetExport : eth0 is up.
2017-12-09 04:10:51,000 INFO (floatip) check wsInterface : eth0:oms, wsFloatIp: XXX.XXX.XXX.XXX.
2017-12-09 04:10:51,000 INFO (floatip) check omInterface : eth0:oms, omFloatIp: XXX.XXX.XXX.XXX.
2017-12-09 04:10:51,000 INFO (floatip) check  wsFloatIp : XXX.XXX.XXX.XXX is reachable.
2017-12-09 04:10:52,000 INFO (floatip) check  omFloatIp : XXX.XXX.XXX.XXX is reachable.
```

- If yes, go to **Step 12**.
- If no, go to **Step 9**.

Step 9 Check whether the omNetExport log is printed after the wsNetExport is detected or whether the interval for printing two logs exceeds 10 seconds or longer.

- If yes, go to **Step 10**.
- If no, go to **Step 12**.

Step 10 View the /var/log/message file of the operating system. For Red hat, check whether sssd is frequently restarted; for SUSE, check whether nscd exception exists.

For example, see whether there is the exception **Can't contact LDAP server**.

sssd restart example:

```
Feb  7 11:38:16 10-132-190-105 sssd[pam]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd[nss]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd[nss]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd[be[default]]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd: Starting up
Feb  7 11:38:16 10-132-190-105 sssd[be[default]]: Starting up
Feb  7 11:38:16 10-132-190-105 sssd[nss]: Starting up
Feb  7 11:38:16 10-132-190-105 sssd[pam]: Starting up
```

nscd exception example:

```
Feb 11 11:44:42 10-120-205-33 nscd: nss_ldap: failed to bind to LDAP server ldaps://10.120.205.55:21780:
Can't contact LDAP server
Feb 11 11:44:43 10-120-205-33 ntpq: nss_ldap: failed to bind to LDAP server ldaps://10.120.205.55:21780:
Can't contact LDAP server
Feb 11 11:44:44 10-120-205-33 ntpq: nss_ldap: failed to bind to LDAP server ldaps://10.120.205.92:21780:
Can't contact LDAP server
```

- If yes, go to **Step 11**.
- If no, go to **Step 12**.

Step 11 Check whether the ldapserver node is faulty, for example, the service IP address is unreachable or the network latency is too long. If the fault lasts periodically, locate and eliminate it and run the **top** command to check whether abnormal software exists.

Collect fault information.

Step 12 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 13 Select the following nodes from the **Service** and click **OK**:

- NodeAgent
- Controller
- OS

Step 14 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 15 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.5 ALM-12007 Process Fault

Description

This alarm is generated when the process health check module detects that the process connection status is **Bad** for three consecutive times. The process health check module checks the process status every 5 seconds.

This alarm is cleared when the process can be connected.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The service provided by the process is unavailable.

Possible Causes

- The instance process is abnormal.
- The disk space is insufficient.

NOTE

If a large number of process fault alarms exist in a time segment, files in the installation directory may be deleted mistakenly or permission on the directory may be modified.

Procedure

Check whether the instance process is abnormal.

Step 1 In the FusionInsight Manager portal, click **O&M > Alarm > Alarms**, click ⌄ in the row where the alarm is located , and click the host name to view the host address for which the alarm is generated

Step 2 On the **Alarms** page, check whether the **ALM-12006 Node Fault** is generated.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Handle the alarm according to **ALM-12006 Node Fault**.

Step 4 Log in to the host for which the alarm is generated as user **root**. Check whether the installation directory user, user group, and permission of the alarm role are correct. The user, user group, and the permission must be **omm:ficommon 750**.

For example, the NameNode installation directory is *${BIGDATA_HOME}*/**FusionInsight_Current/***1_8_NameNode*/**etc**.

- If yes, go to **Step 6**.
- If no, go to **Step 5**.

Step 5 Run the following command to set the permission to **750** and **User:Group** to **omm:ficommon**:

chmod 750 *<folder_name>*

chown omm:ficommon *<folder_name>*

Step 6 Wait for 5 minutes. In the alarm list, check whether **ALM-12007 Process Fault** is cleared.

- If yes, no further action is required.

- If no, go to **Step 7**.

Check whether disk space is sufficient.

Step 7 On the FusionInsight Manager, check whether the alarm list contains **ALM-12017 Insufficient Disk Capacity**.

- If yes, go to **Step 8**.
- If no, go to **Step 11**.

Step 8 Rectify the fault by following the steps provided in **ALM-12017 Insufficient Disk Capacity**.

Step 9 Wait for 5 minutes. In the alarm list, check whether **ALM-12017 Insufficient Disk Capacity** is cleared.

- If yes, go to **Step 10**.
- If no, go to **Step 11**.

Step 10 Wait for 5 minutes. In the alarm list, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 12 According to the service name obtained in **Step 1**, select the component and **NodeAgent** from the **Service** and click **OK**.

Step 13 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.6 ALM-12010 Manager Heartbeat Interruption Between the Active and Standby Nodes

Description

This alarm is generated when the active Mager does not receive the heartbeat signal from the standby Manager within 7 seconds.

This alarm is cleared when the active Manager receives heartbeat signals from the standby Manager.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12010	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

When the active Manager process is abnormal, an active/standby failover cannot be performed, and services are affected.

Possible Causes

- The link between the active and standby Manager is abnormal.
- The node name configuration is incorrect.
- The port is disabled by the firewall.

Procedure

Check whether the network between the active and standby Manager server is normal.

Step 1 In the FusionInsight Manager portal, click **O&M > Alarm > Alarms**, click ⌄ in the row containing the alarm and view the IP address of the standby Manager (Peer Manager) server in the alarm details.

Step 2 Log in to the active Manager server as user **root**.

Step 3 Run the **ping** *standby Manager heartbeat IP address* command to check whether the standby Manager server is reachable.

- If yes, go to **Step 6**.
- If no, go to **Step 4**.

Step 4 Contact the network administrator to check whether the network is faulty.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Rectify the network fault and check whether the alarm is cleared from the alarm list.

- If yes, no further action is required.
- If no, go to **Step 6**.

Step 6 Run the following command to go to the software installation directory:

cd /opt

Step 7 Run the following command to find the configuration file directory of the active and standby nodes.

find -name hacom_local.xml

Step 8 Run the following command to go to the **workspace** directory:

cd${BIGDATA_HOME}/om-server/OMS/workspace0/ha/local/hacom/conf/

Step 9 Run the **vim** command to open the **hacom_local.xml** file. Check whether the local and peer nodes are correctly configured. The local node is configured as the active node, and the peer node is configured as the standby node.

- If yes, go to **Step 12**.
- If no, go to **Step 10**.

Step 10 Modify the configuration of the active and standby nodes in the **hacom_local.xml** file and press **Esc** to return to the command mode. Run the **:wq** command to save the modification and exit.

Step 11 Check whether the alarm is cleared automatically.

- If yes, no further action is required.
- If no, go to **Step 12**.

Check whether the port is disabled by the firewall.

Step 12 Run the **lsof -i :20012** command to check whether the heartbeat ports of the active and standby nodes are enabled. If the command output is displayed, the ports are enabled. Otherwise, the ports are disabled by the firewall.

- If yes, go to **Step 13**.
- If no, go to **Step 16**.

Step 13 Run the **iptables -P INPUT ACCEPT** command to avoid the server disconnection.

Step 14 Run the following command to clear the firewall:

iptables -F

Step 15 Check whether the alarm is cleared from the alarm list.

- If yes, no further action is required.
- If no, go to **Step 16**.

Collect fault information.

Step 16 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 17 Select the following nodes from the **Service** and click **OK**:

- OmmServer
- Controller
- NodeAgent

Step 18 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 19 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.7 ALM-12011 Manager Data Synchronization Exception Between the Active and Standby Nodes

Description

The system checks data synchronization between the active and standby Manager nodes every 60 seconds. This alarm is generated when the standby Manager fails to synchronize files with the active Manager.

This alarm is cleared when the standby Manager synchronizes files with the active Manager.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12011	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Some configurations will be lost after an active/standby switchover because the configuration files on the standby Manager are not updated. Maybe Manager and some components cannot run properly.

Possible Causes

- The link between the active and standby Managers is interrupted or The storage space of the **/srv/BigData/LocalBackup** directory is full.
- The synchronization file does not exist or the file permission is incorrect.

Procedure

Check whether the network between the active Manager server and the standby Manager server is normal.

Step 1 In the FusionInsight Manager portal, click **O&M > Alarm > Alarms**, click ⌄ in the row where the alarm is located and obtain the standby Manager server IP address (Peer Manager IP address) in the alarm details.

Step 2 Log in to the active Manager server as user **root**.

Step 3 Run the **ping** *standby Manager IP address* command to check whether the standby Manager server is reachable.

- If yes, go to **Step 6**.
- If no, go to **Step 4**.

Step 4 Contact the network administrator to check whether the network is faulty.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Rectify the network fault and check whether the alarm is cleared from the alarm list.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check whether the storage space of the /srv/BigData/LocalBackup directory is full.

Step 6 Run the following command to check whether the storage space of the **/srv/BigData/LocalBackup** directory is full:

df -hl /srv/BigData/LocalBackup

- If yes, go to **Step 7**.

- If no, go to **Step 10**.

Step 7 Run the following command to clear unnecessary backup files:

rm -rf *Directory to be cleared*

Example:

rm -rf /srv/BigData/LocalBackup/0/default-oms_20191211143443

Step 8 On FusionInsight Manager, choose **O&M** > **Backup and Restoration** > **Backup Management**.

In the **Operation** column of the backup task to be performed, click **Configure** and change the value of **Maximum Number of Backup Copies** to reduce the number of backup file sets.

Step 9 Wait about 1 minute and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

Check whether the synchronization file exists and whether the file permission is normal.

Step 10 Run the following command to check whether the synchronization file exists.

find /srv/BigData/ -name "sed*"

find /opt -name "sed*"

- If yes, go to **Step 11**.
- If no, go to **Step 12**.

Step 11 Run the following command to view the synchronization file information and permission obtained in **Step 10**.

ll *path of the file to be found*

- If the size of the file is 0 and the permission column is **-**, the file is a junk file. Run the following command to delete it.

  rm -rf *files to be deleted*

  Wait for several minutes and check whether the alarm is cleared. If the alarm persists, go to **Step 12**.
- If the file size is not 0, go to **Step 12**.

Step 12 View the log files generated when the alarm is generated.

1. Run the following command to switch to the HA run log file path.

   cd /var/log/Bigdata/omm/oms/ha/runlog/
2. Decompress and view the log files generated when the alarm is generated.

   For example, if the name of the file to be viewed is **ha.log.2021-03-22_12-00-07.gz**, run the following command:

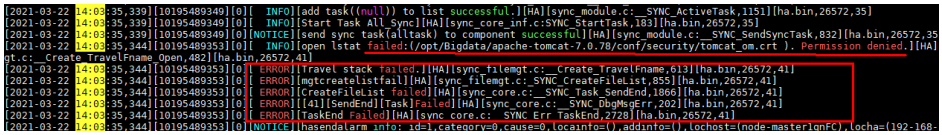
   gunzip *ha.log.2021-03-22_12-00-07.gz*

   vi *ha.log.2021-03-22_12-00-07*

   Check whether error information is reported before and after the alarm generation time.

- If yes, rectify the fault based on the error information. Then go to **Step 13**.

  For example, if the following error information is displayed, the directory permission is insufficient. In this case, change the directory permission to be the same as that on the normal node.



- If no, go to **Step 14**.

Step 13 Wait about 10 minute and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Collect fault information.

Step 14 On the FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 15 Select the following nodes from the **Service** and click **OK**:

- OmmServer
- Controller
- NodeAgent

Step 16 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 17 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.8 ALM-12014 Partition Lost

Description

The system checks the partition status every 60 seconds. This alarm is generated when the system detects that a partition to which service directories are mounted is lost (because the device is removed or goes offline, or the partition is deleted). The system checks the partition status periodically.

This alarm must be manually cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12014	Major	No

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
DirName	Specifies the directory for which the alarm is generated.
PartitionName	Specifies the device partition for which the alarm is generated.

Impact on the System

Service data fails to be written into the partition, and the service system runs abnormally.

Possible Causes

- The hard disk is removed.
- The hard disk is offline, or a bad sector exists on the hard disk.

Procedure

Step 1 On FusionInsight Manager, click **O&M > Alarm > Alarms**, and click ⌄ in the row where the alarm is located.

Step 2 Obtain **HostName**, **PartitionName** and **DirName** from **Location**.

Step 3 Check whether the disk of **PartitionName** on **HostName** is inserted to the correct server slot.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Contact hardware engineers to remove the faulty disk.

Step 5 Log in to the **HostName** node where an alarm is reported and check whether there is a line containing **DirName** in the **/etc/fstab** file as user **root**.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 6 Run the **vi /etc/fstab** command to edit the file and delete the line containing **DirName**.

Step 7 Contact hardware engineers to insert a new disk. For details, see the hardware product document of the relevant model. If the faulty disk is in a RAID group, configure the RAID group. For details, see the configuration methods of the relevant RAID controller card.

Step 8 Wait 20 to 30 minutes (The disk size determines the waiting time), and run the **mount** command to check whether the disk has been mounted to the **DirName** directory.

- If yes, manually clear the alarm. No further operation is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 10 Select the **OmmServer** from the Services drop-down list and click **OK**.

Step 11 Set Start Date for log collection to 10 minutes ahead of the alarm generation time and End Date to 10 minutes behind the alarm generation time and click **Download**.

Step 12 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system does not automatically clear this alarm, and you need to manually clear the alarm.

Related Information

None

10.13.9 ALM-12015 Partition Filesystem Readonly

Description

The system checks the partition status every 60 seconds. This alarm is generated when the system detects that a partition to which service directories are mounted enters the read-only mode (due to a bad sector or a faulty file system). The system checks the partition status periodically.

This alarm is cleared when the system detects that the partition to which service directories are mounted exits from the read-only mode (because the file system is restored to read/write mode, the device is removed, or the device is formatted).

Attribute

Alarm ID	Alarm Severity	Auto Clear
12015	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
DirName	Specifies the directory for which the alarm is generated.
PartitionName	Specifies the device partition for which the alarm is generated.

Impact on the System

Service data fails to be written into the partition, and the service system runs abnormally.

Possible Causes

The hard disk is faulty, for example, a bad sector exists.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**, click ⌄ in the row where the alarm is located.

Step 2 Obtain **HostName** and **PartitionName** from **Location**. **HostName** is the node where the alarm is reported, and **PartitionName** is the partition of the faulty disk.

Step 3 Contact hardware engineers to check whether the disk is faulty. If the disk is faulty, remove it from the server.

Step 4 After the disk is removed, alarm **ALM-12014 Partition Lost** is reported. Handle the alarm. For details, see **ALM-12014 Partition Lost**. After the alarm **ALM-12014**

Partition Lost is cleared, alarm **ALM-12015 Partition Filesystem Readonly** is automatically cleared.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.10 ALM-12016 CPU Usage Exceeds the Threshold

Description

The system checks the CPU usage every 30 seconds and compares the actual CPU usage with the threshold. The CPU usage has a default threshold. This alarm is generated when the CPU usage exceeds the threshold for several times (configurable, 10 times by default) consecutively.

The alarm is cleared in the following two scenarios: The value of **Trigger Count** is 1 and the CPU usage is smaller than or equal to the threshold; the value of **Trigger Count** is greater than 1 and the CPU usage is smaller than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12016	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Service processes respond slowly or become unavailable.

Possible Causes

- The alarm threshold or alarm smoothing times are incorrect.
- CPU configuration cannot meet service requirements. The CPU usage reaches the upper limit.

Procedure

Check whether the alarm threshold or alarm Trigger Count are correct.

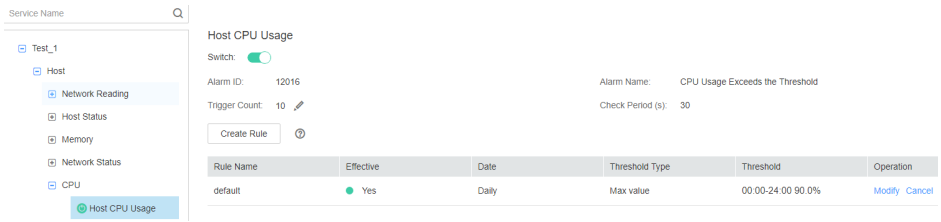
Step 1 Change the alarm threshold and alarm **Trigger Count** based on CPU usage.

On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds >** *Name of the desired cluster* > **Host** > **CPU** > **Host CPU Usage** and change the alarm smoothing times based on CPU usage, as shown in **Figure 10-22**.

NOTE

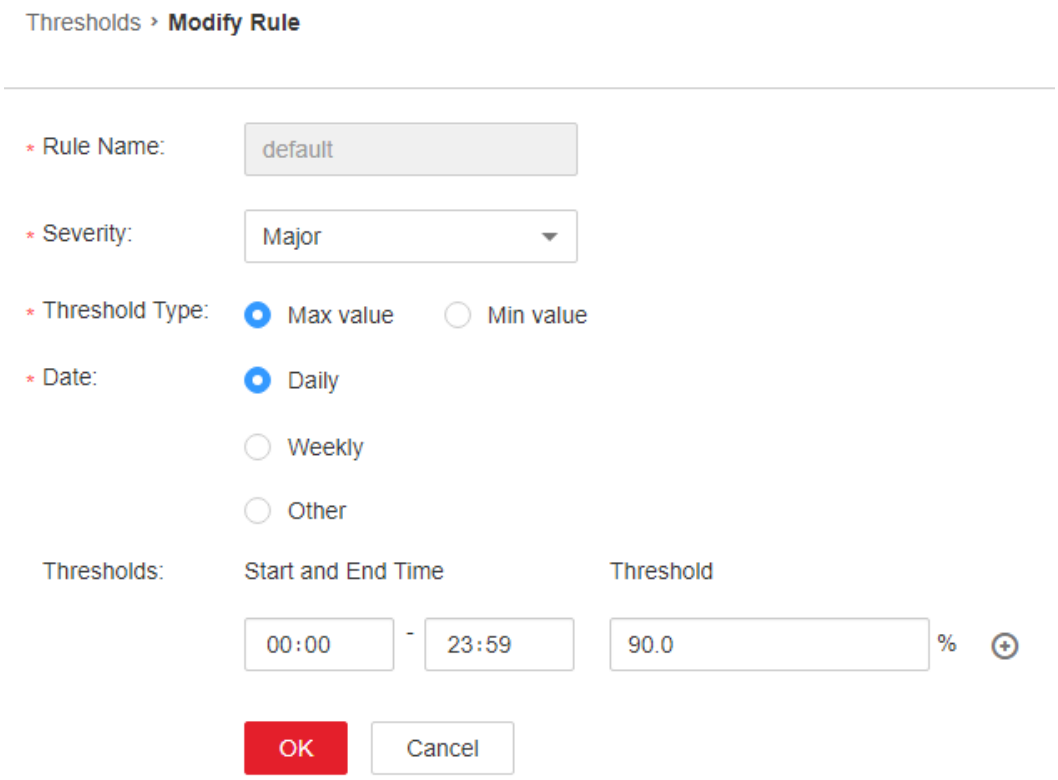
This option defines the alarm check phase. **Trigger Count** indicates the alarm check threshold. An alarm is generated when the number of check times exceeds the threshold.

Figure 10-22 Setting alarm smoothing times



On **Host CPU Usage** page and click **Modify** in the **Operation** column to change the alarm threshold, as shown in **Figure 10-23**.

Figure 10-23 Setting an alarm threshold



Step 2 After 2 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Check whether the CPU usage reaches the upper limit.

Step 3 In the alarm list on FusionInsight Manager, click ⌄ in the row where the alarm is located to view the alarm host address in the alarm details.

Step 4 On the **Hosts** page, click the node on which the alarm is reported.

Step 5 View the CPU usage for 5 minutes. If the CPU usage exceeds the threshold for multiple times, contact the system administrator to add more CPUs.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager in the active cluster, choose **O&M** > **Log** > **Download**.

Step 8 Select **OmmServer** from the **Service** and click **OK**.

Step 9 Set **Start Date** for log collection to 10 minutes ahead of the alarm generation time and **End Date** to 10 minutes behind the alarm generation time in **Time Range** and click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.11 ALM-12017 Insufficient Disk Capacity

Description

The system checks the host disk usage of the system every 30 seconds and compares the actual disk usage with the threshold. The disk usage has a default threshold, this alarm is generated when the host disk usage exceeds the specified threshold.

When the **Trigger Count** is 1, this alarm is cleared when the usage of a host disk partition is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the usage of a host disk partition is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12017	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
PartitionName	Specifies the device partition for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Service processes become unavailable.

Possible Causes

- The alarm threshold is incorrect.
- Disk configuration of the server cannot meet service requirements.

Procedure

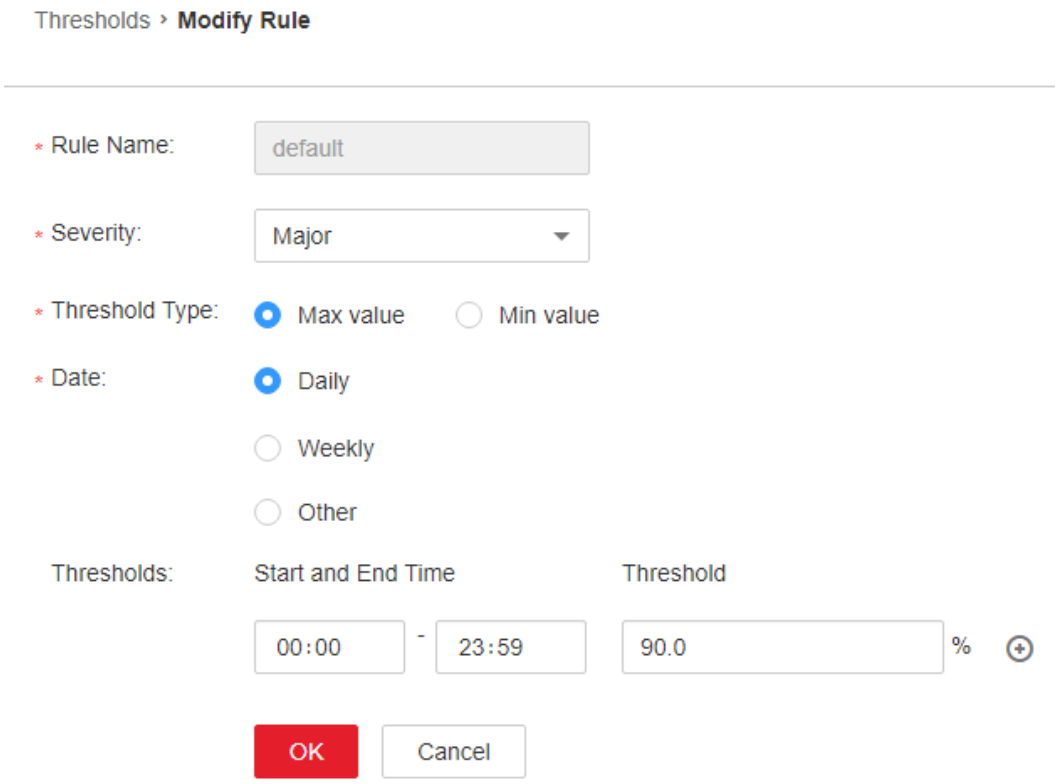
Check whether the alarm threshold is appropriate.

Step 1 Log in to FusionInsight Manager, choose **O&M** > **Alarm > Thresholds >** *Name of the desired cluster* > **Host** > **Disk** > **Disk Usage** and check whether the threshold (configurable, 90% by default) is appropriate.

- If yes, go to **Step 2**.
- If no, go to **Step 4**.

Step 2 Choose **O&M** > **Alarm > Thresholds >** *Name of the desired cluster* > **Host** > **Disk** > **Disk Usage** and click **Modify** in the **Operation** column to change the alarm threshold based on site requirements. As shown in **Figure 10-24**:

Figure 10-24 Setting an alarm threshold



Step 3 After 2 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the disk usage reaches the upper limit.

Step 4 In the alarm list on FusionInsight Manager, click ⌄ in the row where the alarm is located to view the alarm host name and disk partition information in the alarm details.

Step 5 Log in to the node where the alarm is generated as user **root**.

Step 6 Run the **df -lmPT | awk '$2 != "iso9660"' | grep '^/dev/' | awk '{"readlink -m "$1 | getline real }{$1=real; print $0}' | sort -u -k 1,1** command to check the system disk partition usage. Check whether the disk is mounted to the following directories based on the disk partition name obtained in **Step 4**: **/**, **/opt**, **/tmp**, **/var**, **/var/log**, and **/srv/BigData**(can be customized).

- If yes, the disk is a system disk. Then go to **Step 10**.
- If no, the disk is not a system disk. Then go to **Step 7**.

Step 7 Run the **df -lmPT | awk '$2 != "iso9660"' | grep '^/dev/' | awk '{"readlink -m "$1 | getline real }{$1=real; print $0}' | sort -u -k 1,1** command to check the system disk partition usage. Determine the role of the disk based on the disk partition name obtained in **Step 4**.

Step 8 Check the disk service.

In MRS, check whether the disk service is HDFS, Yarn, Kafka, Supervisor.

- If yes, adjust the capacity. Then go to **Step 9**.
- If no, go to **Step 12**.

Step 9 After 2 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Step 10 Run the **find / -xdev -size +500M -execls -l {} \;** command to check whether a file larger than 500 MB exists on the node and disk.

- If yes, go to **Step 11**.
- If no, go to **Step 12**.

Step 11 Handle the large file and check whether the alarm is cleared 2 minutes later.

- If yes, no further action is required.
- If no, go to **Step 12**.

Step 12 Contact the system administrator to expand the disk capacity.

Step 13 After 2 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Collect fault information.

Step 14 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 15 Select **OMS** from the **Service** and click **OK**.

Step 16 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 17 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.12 ALM-12018 Memory Usage Exceeds the Threshold

Description

The system checks the memory usage of the system every 30 seconds and compares the actual memory usage with the threshold. The memory usage has a default threshold, this alarm is generated when the value of the memory usage exceeds the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the host memory usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1,

this alarm is cleared when the host memory usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12018	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Service processes respond slowly or become unavailable.

Possible Causes

- Memory configuration cannot meet service requirements. The memory usage reaches the upper limit.
- The SUSE 12.X OS has an earlier **free** command. The calculated memory usage cannot reflect the real-world memory usage.

Procedure

Perform the following operations if SUSE 12.X is used.

Step 1 Log in to any node in the cluster as user **root**, and run the **cat /etc/*-release** command to check whether the OS is SUSE 12.X as user **root**.

- If yes, go to **Step 2**.
- If no, go to **Step 4**.

Step 2 Run the **cat /proc/meminfo | grep Mem** command to check the real-world memory usage of the OS.

```
MemTotal: 263576192 kB
MemFree: 198283116 kB
MemAvailable: 227641452 kB
```

Step 3 Calculate the real-world memory usage: Memory usage = 1 - (Memory available/Memory total)

- If the memory usage is lower than 90%, manually disable transferring from monitoring indicators to alarms.
- If the memory usage is higher than 90%, go to **Step 4**.

Expand the system.

Step 4 In the alarm list on FusionInsight Manager, click ⌄ in the row where the alarm is located to view the alarm host address in the alarm details.

Step 5 Log in to the host where the alarm is generated as user **root**.

Step 6 If the memory usage exceeds the threshold, perform memory capacity expansion.

Step 7 Run the command **free -m | grep Mem\: | awk '{printf("%s,", ($3-$6-$7) * 100 / $2)}'** to check the system memory usage.

Step 8 Wait for 5 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager in the active cluster, choose **O&M** > **Log** > **Download**.

Step 10 Select **OmmServer** from the **Service** and click **OK**.

Step 11 Click ✎ in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.13 ALM-12027 Host PID Usage Exceeds the Threshold

Description

The system checks the PID usage every 30 seconds and compares the actual PID usage with the default PID usage threshold. This alarm is generated when the system detects that the PID usage exceeds the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the PID usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the PID usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12027	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

No PID is available for new processes and service processes are unavailable.

Possible Causes

Too many processes are running on the node. You need to increase the value of **pid_max**.

Procedure

Increase the value of pid_max.

Step 1 In the alarm list on FusionInsight Manager, click ⌄ in the row where the alarm is located to view the alarm host address in the alarm details.

Step 2 Log in to the host where the alarm is generated as user **root**.

Step 3 Run the **cat /proc/sys/kernel/pid_max**command to check the value of **pid_max**.

Step 4 If the PID usage exceeds the threshold, run the command **echo** *new value* **> /proc/sys/kernel/pid_max** to enlarge the value of **pid_max**.

Example: **echo 65536 > /proc/sys/kernel/pid_max**

Step 5 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log > Download**.

Step 7 Select all services from the **Service** and click **OK**.

Step 8 Click ✎ in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.14 ALM-12028 The number of processes that are in the D state on the host exceeds the threshold

Description

The system periodically checks the number of D state processes of user **omm** on the host every 30 seconds and compares the number with the threshold. The number of host D state processes has a default threshold, and this alarm is generated when the number of processes exceeds the specified threshold.

When the **Trigger Count** is 1, this alarm is cleared when the number is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the number is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12028	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Excessive system resources are used and the service process responds slowly.

Possible Causes

The host responds slowly to I/O (disk I/O and network I/O) requests and a process is in the D state.

Procedure

Check the process that is in the D state.

Step 1 In the alarm list on FusionInsight Manager, click ⌄ in the row where the alarm is located to view the alarm host address in the alarm details.

Step 2 Log in to the alarm host as user **root**, and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to view the PID of the process of user **omm** that is in the D state:

ps -elf | grep -v "\[thread_checkio\]" | awk 'NR!=1 {print $2, $3, $4}' | grep omm | awk -F' ' '{print $1, $3}' | grep D | awk '{print $2}'

Step 4 Check whether no command output is displayed in **Step 3**.

- If yes, the service process is running properly and no further action is required, go to **Step 6**.
- If no, go to **Step 5**.

Step 5 Switch to user **root**, Run the **reboot** command to restart the alarm host. (Restarting the host brings certain risks. Ensure that the service process runs properly after the restart).

Step 6 Wait for 5 minutes. Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 8 Select **OMS** from the **Service** and click **OK**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.15 ALM-12033 Slow Disk Fault

Description

The system runs the **iostat** command every 3 seconds to monitor the system indicator of disk I/O. If the **svctm** value is greater than 100 ms and greater than 1.5 times the **svctm_average** value within 300 seconds, it is considered as a slow period. If the number of slow periods within 300s is greater than 50%, the system considers that the disk is faulty and reports an alarm.

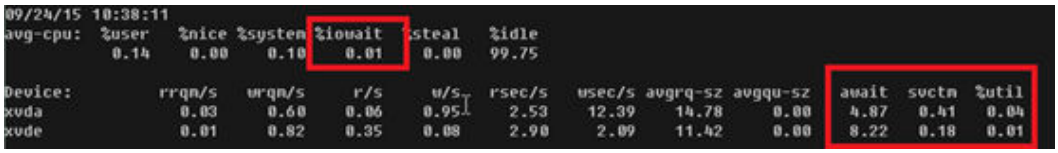
NOTE

The value of **svctm_average** is the average value of all disk **svctm** on the current node.

This alarm is automatically cleared after the disk is replaced.

The alarm detecting principle is as follows:

On the Linux platform, run the **iostat-x -t 1** command to check whether the I/O is faulty. Specifically, check values of parameters in the red box in the following figure.



- %iowait: Specifies the percentage of the time when the CPU waits for I/O to the entire CPU period. If the value exceeds 50% or is significantly greater than the value of **%system**, **%user**, and **%idle**, the I/O may be faulty.
- await: Specifies the sum of the disk I/O waiting time and I/O service time. The value of this parameter does not exceed 20. The value of this parameter for other DataNode disks can be slightly higher but cannot exceed 40.
- svctm: Specifies the time when the I/O service of the disk is changed.
- %util: Specifies the busy degree of the disk. If the value exceeds 80%, the disk maybe busy.

If the value of **%util** is greater than 10 and the value of **svctm** is greater than 100, the I/O is recorded as faulty. This alarm is generated when the I/O is recorded as faulty for 30 times in the 60 times of checks.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12033	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
DiskName	Specifies the disk for which the alarm is generated.

Impact on the System

Service performance deteriorates and service processing capabilities become poor, and even the service is unavailable.

Possible Causes

The disk is aged or has bad sectors.

Procedure

Check the disk status.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms**.

Step 2 View the detailed information about the alarm to obtain the values of the **HostName** and the **DiskName** fields and the information about the faulty disk for which the alarm is generated.

Step 3 Check whether the node for which the alarm is generated is in the virtualization environment.

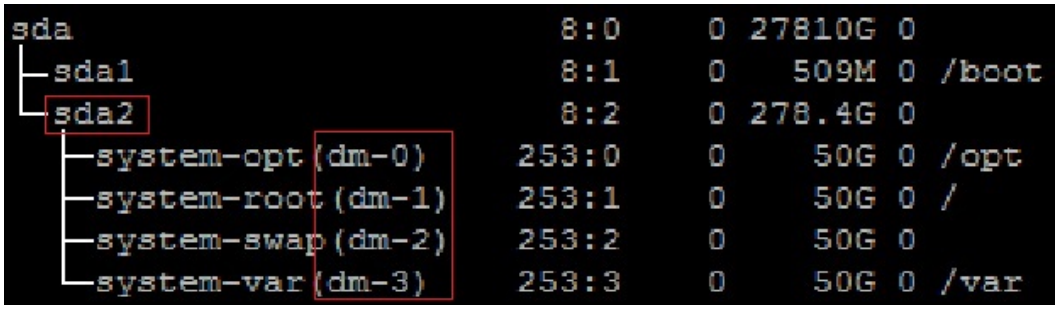
- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 Check whether the storage performance provided by the virtualization environment meets the hardware requirements. Then go to **Step 5** after the check is complete.

Step 5 Log in to the node where the alarm is generated as user **root**. Run the **df -h** command and check whether the command output contains the value of **DiskName**.

- If yes, go to **Step 7**.
- If no, go to **Step 6**.

Step 6 Run the **lsblk** command and check whether you can find out the mapping relationship between the value of **DiskName** and the disks.



- If yes, go to **Step 7**.
- If no, go to **Step 22**.

Step 7 Log in to the node for which the alarm is generated as user **root**. Run the **lsscsi | grep "/dev/sd[x]"** command to view the disk device information and determine whether the disk has been organized into a RAID group.

NOTE

The value of ***dev/sd[x]*** is the faulty disk name obtained in **Step 2**.

For example, run the following command:

lsscsi | grep "/dev/sda"

In the command output, if ATA, SATA, or SAS is displayed in the third line, the disk has not been organized into a RAID group. If other information is displayed, the disk may have been organized into a RAID group.

- If yes, go to **Step 12**.
- If no, go to **Step 8**.

Step 8 Run the **smartctl -i /dev/sd[x]** command to check whether the hardware supports SMART.

For example, run the following command:

smartctl -i /dev/sda

In the command output, if **SMART support is: Enabled** is displayed, the hardware supports SMART. If **Device does not support SMART** is displayed, the hardware does not support SMART.

- If yes, go to **Step 9**.
- If no, go to **Step 17**.

Step 9 Run the **smartctl -H --all /dev/sd[x]** command to check basic SMART information and determine whether the disk is working correctly.

For example, run the following command:

smartctl -H --all /dev/sda

Check **SMART overall-health self-assessment test result** in the command output. If the result is **FAILED**, the disk is faulty and needs to be replaced. If the result is **PASSED**, check the count of **Reallocated_Sector_Ct** or **Elements in grown defect list**. If the count is greater than 100, the disk is faulty and needs to be replaced.

- If yes, go to **Step 10**.
- If no, go to **Step 18**.

Step 10 Run the **smartctl -l error -H /dev/sd[x]** command to check the Glist of the disk and determine whether the disk is working correctly.

For example, run the following command:

smartctl -l error -H /dev/sda

Check the **Command/Featrue_name** column in the command output. If **READ SECTOR(S)** or **WRITE SECTOR(S)** is displayed, the disk has bad sectors. If other errors occur, the disk circuit is faulty. The preceding errors indicate that the disk is abnormal and needs to be replaced.

If **No Errors Logged** is displayed, no error log exists. You can perform step 9 to trigger the disk SMART self-check.

- If yes, go to **Step 11**.
- If no, go to **Step 18**.

Step 11 Run the **smartctl -t long /dev/sd[x]** command to trigger the disk SMART self-check. After the command is executed, the time when the self-check is to be completed is displayed. After the self-check is completed, repeat **Step 9** and **Step 10** to check whether the disk is working properly.

For example, run the following command:

smartctl -t long /dev/sda

- If yes, go to **Step 17**.
- If no, go to **Step 18**.

Step 12 Run the **smartctl -d [sat|scsi]+megaraid,[DID] -H --all /dev/sd[x]** command to check whether the hardware supports SMART.

NOTE

- ***[sat|scsi]*** indicates the disk type. The preceding two types need to be used.
- ***[DID]*** *indicates the slot information. Slots 0 to 15 need to be used.*

For example, run the following commands in sequence:

smartctl -d sat+megaraid,0 -H --all /dev/sda

smartctl -d sat+megaraid,1 -H --all /dev/sda

smartctl -d sat+megaraid,2 -H --all /dev/sda

...

Run the commands that combine different disk types and slots. In a command output, if **SMART support is: Enabled** is displayed, the disk supports SMART. Record the parameters of the disk type and slot combination. If **SMART support is: Enabled** is not displayed in the outputs of all the preceding command combinations, the disk does not support SMART.

- If yes, go to **Step 13**.
- If no, go to **Step 16**.

Step 13 Run the **smartctl -d [sat|scsi]+megaraid,[DID] -H --all /dev/sd[x]** command recorded in **Step 12** to check basic SMART information and determine whether the disk is working correctly.

For example, run the following command:

smartctl -d sat+megaraid,2 -H --all /dev/sda

Check **SMART overall-health self-assessment test result** in the command output. If the result is **FAILED**, the disk is faulty and needs to be replaced. If the result is **PASSED**, check the count of **Reallocated_Sector_Ct** or **Elements in grown defect list**. If the count is greater than 100, the disk is faulty and needs to be replaced.

- If yes, go to **Step 14**.
- If no, go to **Step 18**.

Step 14 Run the **smartctl -d [sat|scsi]+megaraid,[DID] -l error -H /dev/sd[x]** command to check the Glist of the disk and determine whether the disk is working correctly.

For example, run the following command:

smartctl -d sat+megaraid,2 -l error -H /dev/sda

Check the **Command/Featrue_name** column in the command output. If **READ SECTOR(S)** or **WRITE SECTOR(S)** is displayed, the disk has bad sectors. If other errors occur, the disk circuit is faulty. The preceding errors indicate that the disk is abnormal and needs to be replaced.

If **No Errors Logged** is displayed, no error log exists. You can perform step 9 to trigger the disk SMART self-check.

- If yes, go to **Step 15**.
- If no, go to **Step 18**.

Step 15 Run the **smartctl -d [sat|scsi]+megaraid,[DID] -t long /dev/sd[x]** command to trigger the disk SMART self-check. After the command is executed, the time when the self-check is to be completed is displayed. After the self-check is completed, repeat **Step 13** and **Step 14** to check whether the disk is working properly.

For example, run the following command:

smartctl -d sat+megaraid,2 -t long /dev/sda

- If yes, go to **Step 17**.
- If no, go to **Step 18**.

Step 16 If the configured RAID card does not support SMART, the disk usually does not support SMART. In this case, use the check tool provided by the corresponding RAID card vendor to solve the problem. Then go to **Step 17**.

For example, LSI is a MegaCLI tool.

Step 17 On FusionInsight Manager, choose **O&M > Alarm > Alarms**, and click **Clear** in the **Operation** column of the alarm and check whether such alarm is generated for the same disk continuously.

If the alarm is reported for three times for the current disk, you are advised to replace the disk.

- If yes, go to **Step 18**.
- If no, no further action is required.

Replace the disk.

Step 18 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms**.

Step 19 View the detailed information about the alarm to obtain the values of the **HostName** and the **DiskName** fields and the information about the faulty disk for which the alarm is generated.

Step 20 Replace the faulty disk.

Step 21 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 22**.

Collect fault information.

Step 22 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 23 Select **OMS** from the **Service** and click **OK**.

Step 24 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 25 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.16 ALM-12034 Periodical Backup Failure

Description

The system executes the periodic backup task every 60 minutes. This alarm is generated when a periodical backup task fails to be executed. This alarm is cleared when the next backup task is executed successfully.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12034	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
TaskName	Specifies the task.

Impact on the System

There are not available backup packages for a long time, so the system cannot be restored in case of exceptions.

Possible Causes

The alarm cause depends on the task details. Handle the alarm according to the logs and alarm details.

Procedure

Check whether the disk space is sufficient.

Step 1 In the FusionInsight Manager portal, click **O&M > Alarm > Alarms**.

Step 2 In the alarm list, click ⌄ in the row where the alarm is located and obtain **TaskName** from **Location**.

Step 3 Choose **O&M** > **Backup and Restoration > Backup Management**.

Step 4 Search for the backup task based on **TaskName** and click **More** in the **Operation** column. In the displayed dialog box, click **View History** and view the task details.

Step 5 In the displayed dialog box and click ⌄ to check whether the following message is displayed: Failed to backup xx due to insufficient disk space, move the data in the xx directory to other directories.

- If yes, go to **Step 6**.
- If no, go to **Step 13**.

Step 6 Choose **Backup Path** > **View** and obtain the **Backup Path**.

Step 7 Log in to the node as user **root** and run the following command to check the node mounting details:

df -h

Step 8 Check whether the available space of the node to which the backup path is mounted is less than 20 GB.

- If yes, go to **9**.
- If no, go to **Step 13**.

Step 9 Check whether there are many backup packages in the backup directory.

- If yes, go to **Step 10**.
- If no, go to **Step 13**.

Step 10 Enable the available space of the node to which the backup directory is mounted to be greater than 20 GB by moving backup packages out of the backup directory or delete the backup packages.

Step 11 After the problem is resolved, perform the backup task again and check whether the backup task execution is successful.

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 After 2 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collect fault information.

Step 13 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 14 Select **Controller** from the **Service** and click **OK**.

Step 15 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.17 ALM-12035 Unknown Data Status After Recovery Task Failure

Description

After the recovery task fails, the system automatically rolls back every 60 minutes. If the rollback fails, data may be lost. If this occurs, an alarm is reported. This alarm is cleared when the next recovery task execution is successful.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12035	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
TaskName	Specifies the task.

Impact on the System

After the recovery task fails, the system automatically rolls back. If the rollback fails, data may be lost or the data status may be unknown, which may affect services.

Possible Causes

The alarm cause depends on the task details. Handle the alarm according to the logs and alarm details.

Procedure

Collect fault information.

Step 1 In the FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services**, and check whether the running status of the component meets the requirements. (The OMS and DBService must be in the normal state, and other components must be stopped.)

- If yes, go to **Step 9**.
- If no, go to **Step 2**.

Step 2 Restore the component status as required and start the recovery task again.

Step 3 Log in to the FusionInsight Manager portal and click **O&M** > **Alarm** > **Alarms**.

Step 4 In the alarm list, click ⌄ in the row where the alarm is located to obtain **TaskName** from **Location**.

Step 5 Choose **O&M** > **Backup and Restoration** > **Restoration Management**.

Step 6 Find the restoration task by **Task Name** and view the task details.

Step 7 Perform the recovery task again and check whether the recovery task execution is successful.

- If yes, go to **8**.
- If no, go to **9**.

Step 8 After 2 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 10 Select **Controller** from the **Service** and click **OK**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.18 ALM-12038 Monitoring Indicator Dumping Failure

Description

After monitoring indicator dumping is configured on FusionInsight Manager, the system checks the monitoring indicator dumping result at the dumping interval (60 seconds by default). This alarm is generated when the dumping fails.

This alarm is cleared when dumping is successful.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12038	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The upper-layer management system cannot obtain monitoring indicators from the FusionInsight Manager system.

Possible Causes

- The server cannot be connected.
- The save path on the server cannot be accessed.
- The monitoring indicator file fails to be uploaded.

Procedure

Check whether the server connection is normal.

Step 1 Check whether the network between the FusionInsight Manager system and the server is normal.

- If yes, go to **Step 3**.
- If no, go to **Step 2**.

Step 2 Contact the network administrator to recover the network and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Step 3 Choose **System** > **Interconnection** > **Upload Performance Data** and check whether the FTP username, password, port, dump mode, and public key configured on the upload performance data page are consistent with the configuration on the server.

- If yes, go to **Step 5**.
- If no, go to **Step 4**.

Step 4 Enter the correct configuration information, click **OK**, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check the permission of the save path on the server is correct.

Step 5 Choose **System** > **Interconnection** > **Upload Performance Data** and check the configuration items **FTP Username**, **Save Path**, and **Dump Mode**.

- If the dump mode is FTP, go to **Step 6**.
- If the dump mode is SFTP, go to **Step 7**.

Step 6 Log in to the server in FTP mode. In the default path, check whether **FTP Username** has the read and write permission of the relative path **Save Path**.

- If yes, go to **Step 9**.
- If no, go to **Step 8**.

Step 7 Log in to the server in SFTP mode and check whether **FTP Username** has the read and write permission of the absolute path **Save Path**.

- If yes, go to **Step 9**.
- If no, go to **Step 8**.

Step 8 Add the read and write permission and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Check whether the save path on the server has sufficient disk space.

Step 9 Log in to the server and check whether the save path has sufficient disk space.

- If yes, go to **Step 11**.
- If no, go to **Step 10**.

Step 10 Delete unnecessary files or go to the monitoring indicator dumping configuration page to change the save path. Then, check whether the save path has sufficient disk space.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 12 Select **OMS** from the **Service** and click **OK**.

Step 13 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.19 ALM-12039 Active/Standby OMS Databases Not Synchronized

Description

The system checks the data synchronization status between the active and standby OMS Databases every 10 seconds. This alarm is generated when the synchronization status cannot be queried for 30 consecutive times or when the synchronization status is abnormal.

This alarm is cleared when the data synchronization status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12039	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Local GaussDB HA IP	Specifies the HA IP address of the local GaussDB.
Peer GaussDB HA IP	Specifies the HA IP address of the peer GaussDB.
SYNC_PERCENT	Specifies the synchronization percentage.

Impact on the System

When data is not synchronized between the active and standby OMS Databases, data may be lost or abnormal if the active instance becomes abnormal.

Possible Causes

- The network between the active and standby nodes is unstable.
- The standby OMS Database is abnormal.
- The standby node disk space is full.

Procedure

Check whether the network between the active and standby nodes is normal.

Step 1 Log in to FusionInsight Manager, click **O&M > Alarm > Alarms**, click ⌄ in the row where the alarm is located, and query the standby OMS Database IP address.

Step 2 Log in to the active OMS Database node as user **root**.

Step 3 Run the **ping** *Standby OMS Database heartbeat IP address* command to check whether the standby OMS Database node is reachable.

- If yes, go to **Step 6**.
- If no, go to **Step 4**.

Step 4 Contact the network administrator to check whether the network is faulty.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Rectify the network fault and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check whether the standby OMS Database is normal.

Step 6 Log in to the standby OMS Database node as user **root**.

Step 7 Run the **su - omm** command to switch to user **omm**.

Step 8 Go to the **${BIGDATA_HOME}/om-server/om/sbin/** directory and run the **./status-oms.sh** command to check whether the OMS Database resource status of the standby DBService is normal. In the command output, check whether the following information is displayed in the row where **ResName** is **gaussDB**:

For example:

```
10_10_10_231 gaussDB Standby_normal Normal Active_standby
```

- If yes, go to **Step 9**.
- If no, go to **Step 16**.

Check whether the standby node disk space is full.

Step 9 Log in to the standby OMS Database node as user **root**.

Step 10 Run the **su - omm** command to switch to user **omm**.

Step 11 Run the **echo ${BIGDATA_DATA_HOME}/dbdata_om** command to obtain the OMS Database data directory.

Step 12 Run the **df -h** command to view the system disk partition usage information.

Step 13 Check whether the disk where the OMS Database data directory is mounted is full.

- If yes, go to **Step 14**.
- If no, go to **Step 16**.

Step 14 Expand the disk capacity.

Step 15 After the disk capacity is expanded, wait 2 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 16**.

Collect fault information.

Step 16 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 17 Select **OMMServer** from the **Service** and click **OK**.

Step 18 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 19 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.20 ALM-12040 Insufficient System Entropy

Description

The system checks the entropy at 00:00:00 every day, and performs five consecutive checks each time. First, the system checks whether the rng-tools or haveged tool is enabled and correctly configured. If not, the system checks the current entropy. If the entropy is smaller than 100 in the five checks, this alarm is generated.

If the true random number mode is configured, random numbers are configured in the pseudo-random number mode, or neither the true random number mode nor the pseudo-random number mode is configured but the entropy is greater than or equal to 100 in at least one check among the five checks, this alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12040	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host name for which the alarm is generated.

Impact on the System

System running is affected.

Possible Causes

The haveged service or rngd service is abnormal.

Procedure

Check and manually configure the system entropy.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms**.

Step 2 Obtain the value of the **HostName** field in **Location**.

Step 3 Log in to the node where the alarm is generated as user **root**.

Step 4 Run the **/bin/rpm -qa | grep -w "haveged"** command to check **haveged** installation status. Check whether the command output is empty.

- If yes, go to **Step 7**.
- If no, go to **Step 5**.

Step 5 Run the **/sbin/service haveged status |grep "running"** command, and view the command output.

- If the command is executed successfully, the haveged service is installed and correctly configured and is running properly. Go to **Step 10**.
- If the command is not executed successfully, the haveged service is not running properly. Go to **Step 7**.

Step 6 Run the **/bin/rpm -qa | grep -w "rng-tools"** command to check **rng-tools** installation status. Check whether the command output is empty.

- If yes, go to **Step 8**.
- If no, go to **Step 7**.

Step 7 Run the **ps -ef | grep -v "grep" | grep rngd | tr-d " " | grep "\-o/dev/random" | grep "\-r/dev/urandom"** command, and view the command output.

- If the command is executed successfully, the rngd service is installed and correctly configured and is running properly. Go to **Step 10**.
- If the command is not executed successfully, the rngd service is not running properly. Go to **Step 8**.

Step 8 Manually configure the system entropy. For details, see **Related Information**.

Step 9 Wait until 00:00:00 on the next day when the system checks the entropy again. Check whether the alarm is cleared automatically.

- If yes, no further action is required.
- If no, go to **Step 10**.

Collect fault information.

Step 10 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 11 Select **NodeAgent** from the **Service** and click **OK**.

Step 12 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

Manually check the system entropy.

Log in to the node as user **root** and run the **cat/proc/sys/kernel/random/entropy_avail** command to check whether the system entropy meets the cluster installation requirements (the entropy must be greater than or equal to 500). If the system entropy is smaller than 500, you can reset it by using one of the following methods:

- Using the haveged tool (true random number mode): Contact the OS provider to install the tool and then start it.
- Using the rng-tools tool (pseudo random number mode): Contact the OS provider to install the tool and then configure the system entropy based on the OS type.
  - In the Red Hat or CentOS environment, run the following commands to configure the system entropy:

    echo 'EXTRAOPTIONS="-r /dev/urandom -o /dev/random -t 1 -i"' >> /etc/sysconfig/rngd

    service rngd start

    chkconfig rngd on
  - In the SUSE environment, run the following commands to configure the system entropy:

    rngd -r /dev/urandom -o /dev/random

    echo "rngd -r /dev/urandom -o /dev/random" >> /etc/rc.d/after.local

10.13.21 ALM-12041 Incorrect Permission on Key Files

Description

The system checks whether the permission, user, and user group information about critical directories or files is normal every 5 minutes. This alarm is generated when the information is abnormal.

This alarm is cleared when the information becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12041	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
PathName	Specifies the path or name of the abnormal file.

Impact on the System

System functions are unavailable.

Possible Causes

The file permission is abnormal or the file is lost due to a user manually modified information such as the file permission, user, and user group, or the system is powered off unexpectedly.

Procedure

Check whether the abnormal file exists and whether the permission on the abnormal file is correct.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**.

Step 2 Check the value of **HostName** to obtain the host name involved in this alarm. Check the value of **PathName** to obtain the path or name of the abnormal file.

Step 3 Log in to the node for which the alarm is generated as user **root**.

Step 4 Run the **ll** *pathName* command, where *pathName* indicates the name of the abnormal file to obtain the user, permission, and user group information about the file or directory.

Step 5 Go to **${BIGDATA_HOME}/om-agent/nodeagent/etc/agent/autocheck** directory. Then run the **vi keyfile** command and search for the name of the abnormal file and check the due permission of the file.

NOTE

To ensure proper configuration synchronization between the active and standby OMS servers, files, directories, and files and sub-directories in the directories configured in **$OMS_RUN_PATH/workspace/ha/module/hasync/plugin/conf/filesync.xml** will also be monitored except files and directories in **keyfile**. User **omm** must have read and write permissions of files and read and execute permissions of directories.

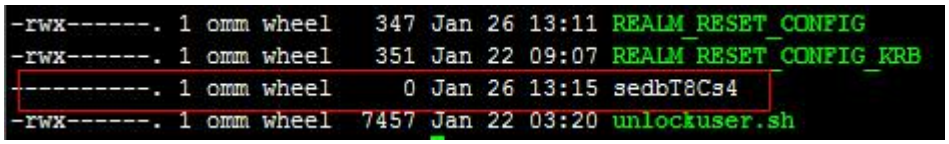
Step 6 Compare the real-world permission of the file with the due permission obtained in **Step 5** and correct the permission, user, and user group information for the file.

Step 7 Wait a hour and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

NOTE

If the disk partition where the cluster installation directory resides is used up, some temporary files will be generated in the program installation directory when running the **sed** command fails. Users do not have the read, write, and execute permissions of these temporary files. The system reports an alarm indicating that permissions of temporary files are abnormal if these files are within the monitoring range of the alarm. Perform the preceding alarm handling processes to clear the alarm. Alternatively, you can directly delete the temporary files after confirming that files with abnormal permissions are temporary. The temporary file generated after a **sed** command execution failure is similar to the following.



Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 9 Select **NodeAgent** from the **Service** and click **OK**.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.22 ALM-12042 Incorrect Configuration of Key Files

Description

The system checks whether critical configurations are correct every 5 minutes. This alarm is generated when the configurations are abnormal.

This alarm is cleared when the configurations become normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12042	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
PathName	Specifies the path or name of the abnormal file.

Impact on the System

Functions related to the file are abnormal.

Possible Causes

The file configuration is modified manually or the system is powered off unexpectedly.

Procedure

Check abnormal file configuration.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**.

Step 2 Check the value of **HostName** to obtain the host name involved in this alarm. Check the value of **PathName** to obtain the path or name of the abnormal file.

Step 3 Log in to the node for which the alarm is generated as user **root**.

Step 4 View the $BIGDATA_LOG_HOME/nodeagent/scriptlog/checkfileconfig.log file and analyze the cause based on the error log. Locate the check standards of the file in the **Related Information** and manually check and modify the file based on the standards.

Run the **vi** *file name* command to enter the editing mode, and then press **Insert** to start editing.

After the modification is complete, press **Esc** to exit the editing mode and enter **:wq** to save the settings and exit.

For example:

vi /etc/ssh/sshd_config

Step 5 Wait a hour and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **NodeAgent** from the **Service** and click **OK**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

- **Check standards of /etc/fstab**

  Check whether the partitions configured in the **/etc/fstab** file can be found in **/proc/mounts**.

  Check whether the swap partitions configured in fstab correspond to those in /proc/swaps.

- **Check the /etc/hosts configuration file.**

  Run **cat /ect/hosts**. If any of the following situations occurs, the **/etc/hosts** configuration file is abnormal:

  a. The **/etc/hosts** file does not exist.

  b. The host name is not configured in the file.

  c. The host name maps to multiple IP addresses in the file.

  d. The IP address corresponding to the host name does not exist in the command output of the **ifconfig** command.

e. One IP address maps to multiple host names in the file.

- **Check standards of /etc/ssh/sshd_config**

  Run the **vi /etc/ssh/sshd_config** command to check whether configuration items are configured as follows:

  a. The value of **UseDNS** must be set to **no**.

  b. The value of **MaxStartups** must be greater than or equal to 1000.

  c. At least one of the **PasswordAuthentication** and **ChallengeResponseAuthentication** parameters must be left blank or at least one of the parameters be set to **yes**.

10.13.23 ALM-12045 Network Read Packet Dropped Rate Exceeds the Threshold

Description

The system checks the network read packet dropped rate every 30 seconds and compares the actual packet dropped rate with the threshold (the default threshold is 0.5%). This alarm is generated when the system detects that the network read packet dropped rate exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds >** *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Dropped Rate**.

When the **Trigger Count** is 1, this alarm is cleared when the network read packet dropped rate is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the network read packet dropped rate is less than or equal to 90% of the threshold.

Alarm detection is disabled by default. If you want to enable this function, check whether alarm sending can be enabled based on section "Check the system environment."

Attribute

Alarm ID	Alarm Severity	Auto Clear
12045	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NetworkCardName	Specifies the network port for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The service performance deteriorates or services time out.

Precautions: In SUSE (kernel: 3.0 or later) or Red Hat 7.2, because the system kernel modifies the mechanism for counting read and discarded packets, this alarm may be generated even when the network is normal. Services are not adversely affected. You are advised to check whether the alarm is caused by this problem based on section "Check the system environment."

Possible Causes

- An OS exception occurs.
- The NIC has configured the active/standby bond mode.
- The alarm threshold is set improperly.
- The cluster network environment is of poor quality.

Procedure

Check the network packet dropped rate.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**, click ⌄ in the row where the alarm is located to view the alarm host name and NIC name in the alarm details.

Step 2 Log in to the node where the alarm is generated as user **omm** and run the **/sbin/ifconfig** *NIC name* command to check whether packet loss occurs on the network.

```
omm@8-5-192-4:~> /sbin/ifconfig eth2
eth2      Link encap:Ethernet  HWaddr E4:35:C8:7B:B5:48
          inet addr:192.168.192.4  Bcast:192.168.255.255  Mask:255.255.0.0
          inet6 addr: fe80::e635:c8ff:fe7b:b548/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5254854 errors:0 dropped:214676 overruns:0 frame:0
          TX packets:329443 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:354839633 (338.4 Mb)  TX bytes:25083094 (23.9 Mb)
```

NOTE

- **IP address**: indicates the value of **HostName** in the alarm location information. To query the value of **OM IP** and **Business IP**, click **Host** on FusionInsight Manager.
- The formula is as follows: Packet loss rate = (Number of dropped packets/Total number of RX packets) x 100%. If the packet loss rate is greater than the system threshold (0.5% by default), packet loss occurs during packet reading on the network.

- If yes, go to **Step 11**.
- If no, go to **Step 3**.

Check the system environment.

Step 3 Log in as user **omm** to the active OMS node or the node for which the alarm is generated.

Step 4 Run the **cat /etc/*-release** command to check the OS type.

- If Red Hat is used, go to **Step 5**.
  ```
  # cat /etc/*-release
  Red Hat Enterprise Linux Server release 7.2 (Santiago)
  ```
- If SUSE is used, go to **Step 6**.
  ```
  # cat /etc/*-release
  SUSE Linux Enterprise Server 11 (x86_64)
  VERSION = 11
  PATCHLEVEL = 3
  ```
- If another OS is used, go to **Step 11**.

Step 5 Run the **cat /etc/redhat-release** command to check whether the OS version is Red Hat 7.2(x86) or Red Hat 7.4(TaiShan).

```
# cat /etc/redhat-release
Red Hat Enterprise Linux Server release 7.2 (Santiago)
```

- If yes, the alarm sending function cannot be enabled. Go to **Step 7**.
- If no, go to **Step 11**.

Step 6 Run the **cat /proc/version** command to check whether the SUSE kernel version is 3.0 or later.

```
# cat /proc/version
Linux version 3.0.101-63-default (geeko@buildhost) (gcc version 4.3.4 [gcc-4_3-branch revision 152973] (SUSE Linux) ) #1 SMP Tue Jun 23 16:02:31 UTC 2015 (4b89d0c)
```

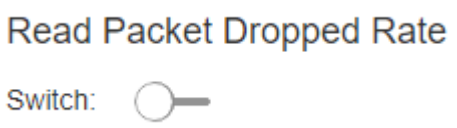
- If yes, the alarm sending function cannot be enabled. Go to **Step 7**.
- If no, go to **Step 11**.

Step 7 Log in to FusionInsight Manager and choose **O&M** > **Alarm** > **Thresholds**.

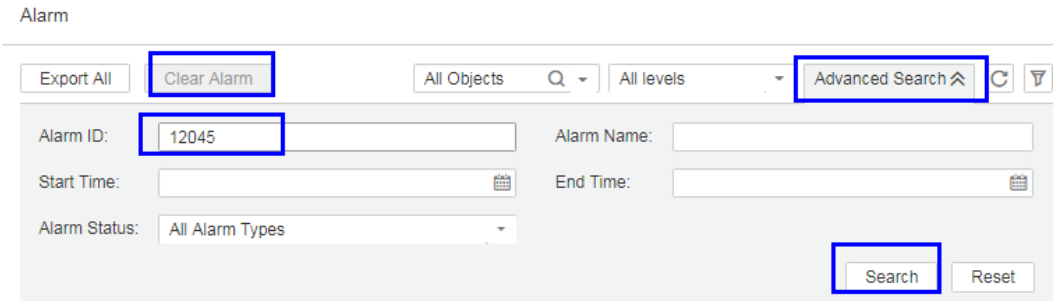
Step 8 In the navigation tree of the **Thresholds** page, choose *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Dropped Rate**. In the area on the right, check whether the **Switch** is on.

- If yes, the alarm sending function has been enabled. Go to **Step 9**.
- If no, the alarm sending function has been disabled. Go to **Step 10**.

Step 9 In the area on the right, close **Switch** to disable the checking of **Network Read Packet Dropped Rate Exceeds the Threshold**. The following figure shows the operation result.



Step 10 On the **Alarm** page of FusionInsight Manager, search for the **12045** alarm. If the alarm is not cleared automatically, clear it manually. No further action is required.



NOTE

The ID of alarm **Network Read Packet Dropped Rate Exceeds the Threshold** is 12045.

Check whether the NIC has configured the active/standby bond mode.

Step 11 Log in to the alarm node as user **omm**. Run the **ls -l /proc/net/bonding** command to check whether directory **/proc/net/bonding** exists on the alarm node.

- If yes, the NIC has configured the active/standby bond mode, as shown in the following. Go to **Step 12**.

```
# ls -l /proc/net/bonding/
total 0
-r--r--r-- 1 root root 0 Oct 11 17:35 bond0
```

- If no, the NIC has not configured the active/standby bond mode, as shown in the following. Go to **Step 14**.

```
# ls -l /proc/net/bonding/
ls: cannot access /proc/net/bonding/: No such file or directory
```

Step 12 Run the **cat /proc/net/bonding/***bond0* command and check whether the value of **Bonding Mode** is **fault-tolerance.**

NOTE

bond0 indicates the name of the bond configuration file. Use the file name queried in **Step 11** in practice.

```
# cat /proc/net/bonding/bond0
Ethernet Channel Bonding Driver: v3.7.1 (April 27, 2011)

Bonding Mode: fault-tolerance (active-backup)
Primary Slave: eth1 (primary_reselect always)
Currently Active Slave: eth1
MII Status: up
MII Polling Interval (ms): 100
Up Delay (ms): 0
Down Delay (ms): 0

Slave Interface: eth0
MII Status: up
Speed: 1000 Mbps
Duplex: full
Link Failure Count: 1
Slave queue ID: 0

Slave Interface: eth1
MII Status: up
Speed: 1000 Mbps
Duplex: full
```

```
Link Failure Count: 1
Slave queue ID: 0
```

- If yes, the NIC has configured the active/standby bond mode. Go to **Step 13**.
- If no, the NIC has not configured the active/standby bond mode. Go to **Step 14**.

Step 13 Check whether the NIC of the **NetworkCardName** parameter is the standby NIC.

- If yes, manually clear the alarm on the Alarms page because the alarm on the standby cannot be automatically cleared. No further action is required.
- If no, go to **Step 14**.

  NOTE

  Method of determining whether an NIC is standby: In the /**proc/net/bonding/***bond0* configuration file, check whether the NIC name of the **NetworkCardName** parameter is the same as the **Slave Interface**, but is different from **Currently Active Slave** (indicating the current active NIC). If the answer is yes, the NIC is a standby one.

Check whether the threshold is set properly.

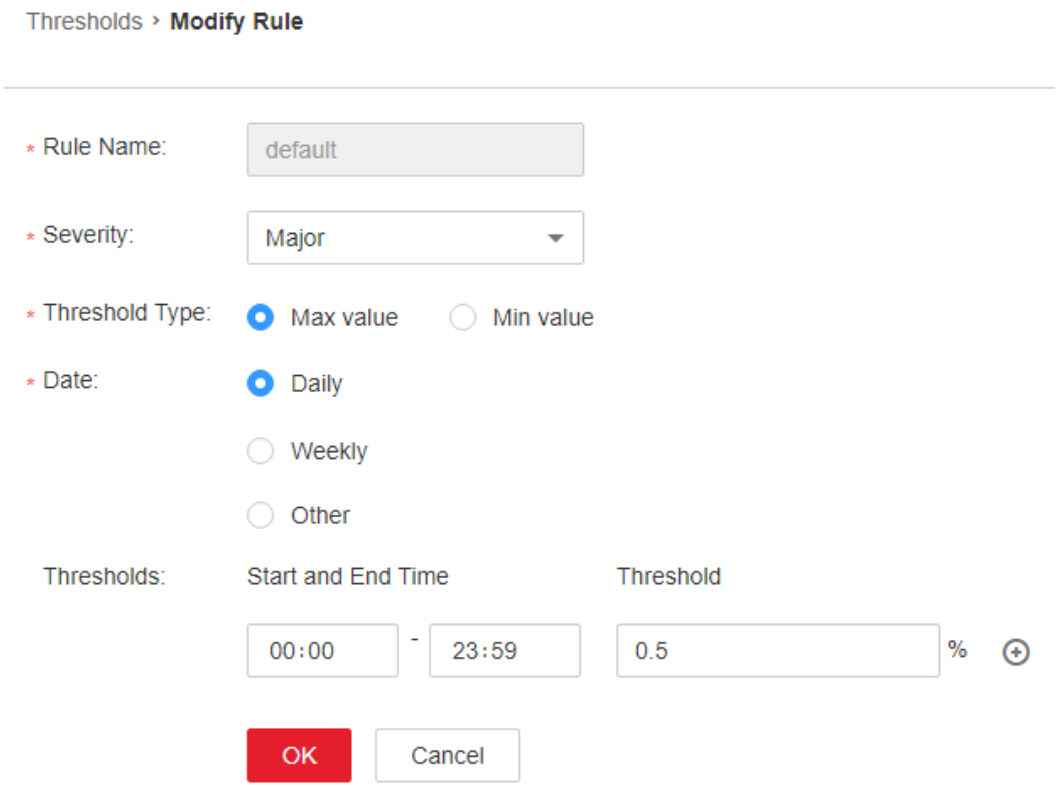
Step 14 Log in to FusionInsight Manager, choose **O&M > Alarm** > **Thresholds >** *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Dropped Rate** and check whether the alarm threshold is set properly. (By default, 0.5% is a proper value. However, users can configure the value as required.)

- If yes, go to **Step 17**.
- If no, go to **Step 15**.

Step 15 Based on actual usage condition, choose **O&M > Alarm** > **Thresholds >** *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Dropped Rate** and click Modify in the Operation column to modify the alarm threshold.

For details, see **Figure 10-25**.

Figure 10-25 Setting alarm thresholds



Step 16 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 17**.

Check whether the network is normal.

Step 17 Contact the system administrator to check whether the network is abnormal.

- If yes, go to **Step 18** to rectify the network fault.
- If no, go to **Step 19**.

Step 18 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 19**.

Collect fault information.

Step 19 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log** > **Download**.

Step 20 Select **OMS** from the **Service** and click **OK**.

Step 21 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 22 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 23 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.24 ALM-12046 Network Write Packet Dropped Rate Exceeds the Threshold

Description

The system checks the network write packet dropped rate every 30 seconds and compares the actual packet dropped rate with the threshold (the default threshold is 0.5%). This alarm is generated when the system detects that the network write packet dropped rate exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds >** *Name of the desired cluster* > **Host** > **Network Writing** > **Write Packet Dropped Rate**.

When the **Trigger Count** is 1, this alarm is cleared when the network write packet dropped rate is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the network write packet dropped rate is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12046	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
NetworkCardName	Specifies the network port for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The service performance deteriorates or services time out.

Possible Causes

- The alarm threshold is set improperly.
- The cluster network environment is of poor quality.

Procedure

Check whether the threshold is set properly.

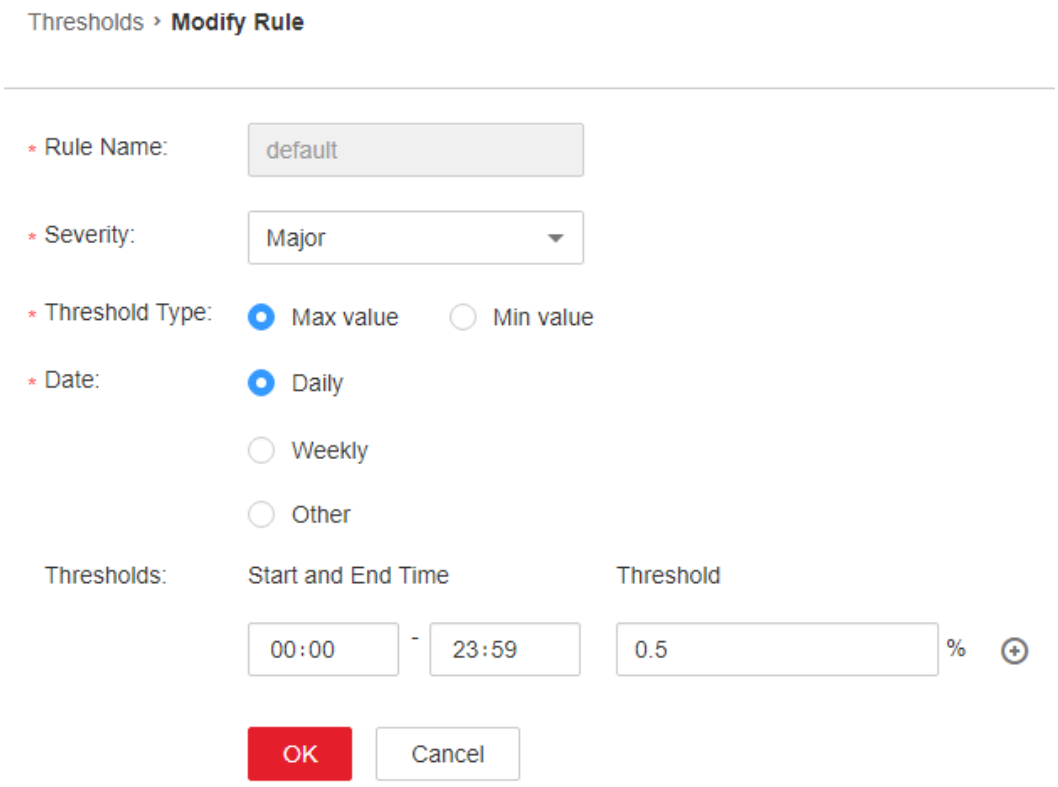
Step 1 On the FusionInsight Manager, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Packet Dropped Rate** to check whether the alarm threshold is set properly. (By default, 0.5% is a proper value. However, users can configure the value as required.)

- If yes, go to **Step 4**.
- If no, go to **Step 2**.

Step 2 Based on actual usage condition, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Packet Dropped Rate** and click **Modify** in the **Operation** column to modify the alarm threshold.

For details, see **Figure 10-26**.

Figure 10-26 Setting alarm thresholds



Step 3 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the network is normal.

Step 4 Contact the system administrator to check whether the network is abnormal.

- If yes, go to **Step 5** to rectify the network fault.
- If no, go to **Step 6**.

Step 5 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log** > **Download**.

Step 7 Select **OMS** from the **Service** and click **OK**.

Step 8 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.25 ALM-12047 Network Read Packet Error Rate Exceeds the Threshold

Description

The system checks the network read packet error rate every 30 seconds and compares the actual packet error rate with the threshold (the default threshold is 0.5%). This alarm is generated when the system detects that the network read packet error rate exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Error Rate**.

When the **Trigger Count** is 1, this alarm is cleared when the network read packet error rate is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the value of the network read packet error rate is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12047	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
NetworkCardName	Specifies the network port for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The communication interrupts intermittently and services time out.

Possible Causes

- The alarm threshold is set improperly.
- The cluster network environment is of poor quality.

Procedure

Check whether the threshold is set properly.

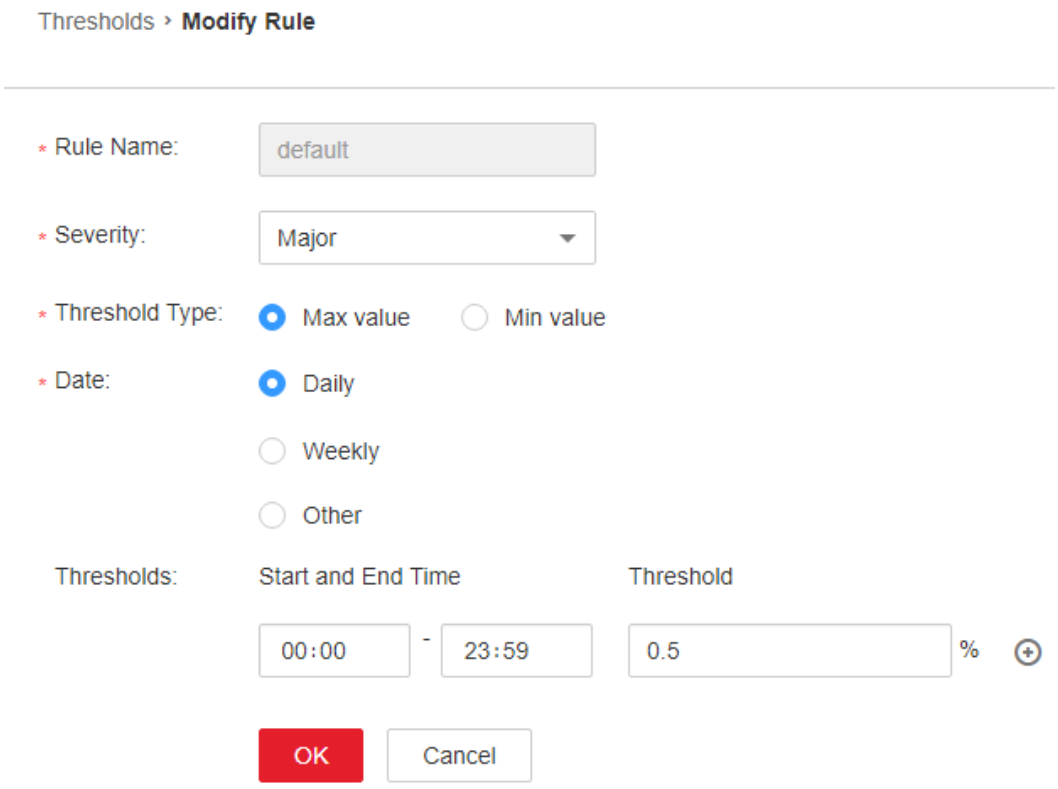
Step 1 On the FusionInsight Manager, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Error Rate** and check whether the alarm threshold is set properly. (By default, 0.5% is a proper value. However, users can configure the value as required.)

- If yes, go to **Step 4**.
- If no, go to **Step 2**.

Step 2 Based on actual usage condition, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Reading** > **Read Packet Error Rate** and click **Modify** in the **Operation** column to modify the alarm threshold.

For details, see **Figure 10-27**.

Figure 10-27 Setting alarm thresholds



Step 3 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the network is normal.

Step 4 Contact the system administrator to check whether the network is abnormal.

- If yes, go to **Step 5** to rectify the network fault.
- If no, go to **Step 6**.

Step 5 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log** > **Download**.

Step 7 Select **OMS** from the **Service** and click **OK**.

Step 8 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.26 ALM-12048 Network Write Packet Error Rate Exceeds the Threshold

Description

The system checks the network write packet error rate every 30 seconds and compares the actual packet error rate with the threshold (the default threshold is 0.5%). This alarm is generated when the system detects that the network write packet error rate exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Packet Error Rate**.

When the **Trigger Count** is 1, this alarm is cleared when the network write packet error rate is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the value of the network write packet error rate is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12048	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
NetworkCardName	Specifies the network port for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The communication interrupts intermittently and services time out.

Possible Causes

- The alarm threshold is set improperly.
- The cluster network environment is of poor quality.

Procedure

Check whether the threshold is set properly.

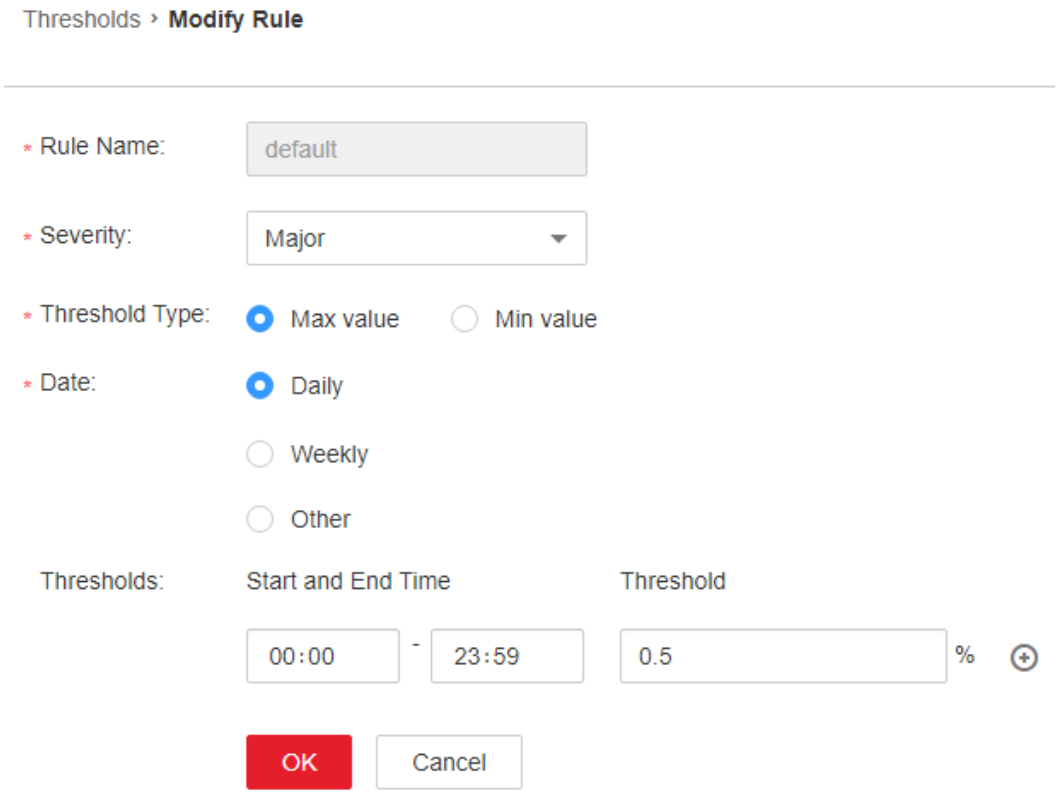
Step 1 On the FusionInsight Manager, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Packet Error Rate** and check whether the alarm threshold is set properly. (By default, 0.5% is a proper value. However, users can configure the value as required.)

- If yes, go to **Step 4**.
- If no, go to **Step 2**.

Step 2 Based on actual usage condition, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Packet Error Rate** and click **Modify** in the **Operation** column to modify the alarm threshold.

For details, see **Figure 10-28**.

Figure 10-28 Setting alarm thresholds



Step 3 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the network is normal.

Step 4 Contact the system administrator to check whether the network is abnormal.

- If yes, go to **Step 5** to rectify the network fault.
- If no, go to **Step 6**.

Step 5 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log** > **Download**.

Step 7 Select **OMS** from the **Service** and click **OK**.

Step 8 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.27 ALM-12049 Network Read Throughput Rate Exceeds the Threshold

Description

The system checks the network read throughput rate every 30 seconds and compares the actual throughput rate with the threshold (the default threshold is 80%). This alarm is generated when the system detects that the network read throughput rate exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Reading** > **Read Throughput Rate**.

When the **Trigger Count** is 1, this alarm is cleared when the network read throughput rate is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the network read throughput rate is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12049	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
NetworkCardName	Specifies the network port for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The service system runs improperly or is unavailable.

Possible Causes

- The alarm threshold is set improperly.
- The network port rate cannot meet the current service requirements.

Procedure

Check whether the threshold is set properly.

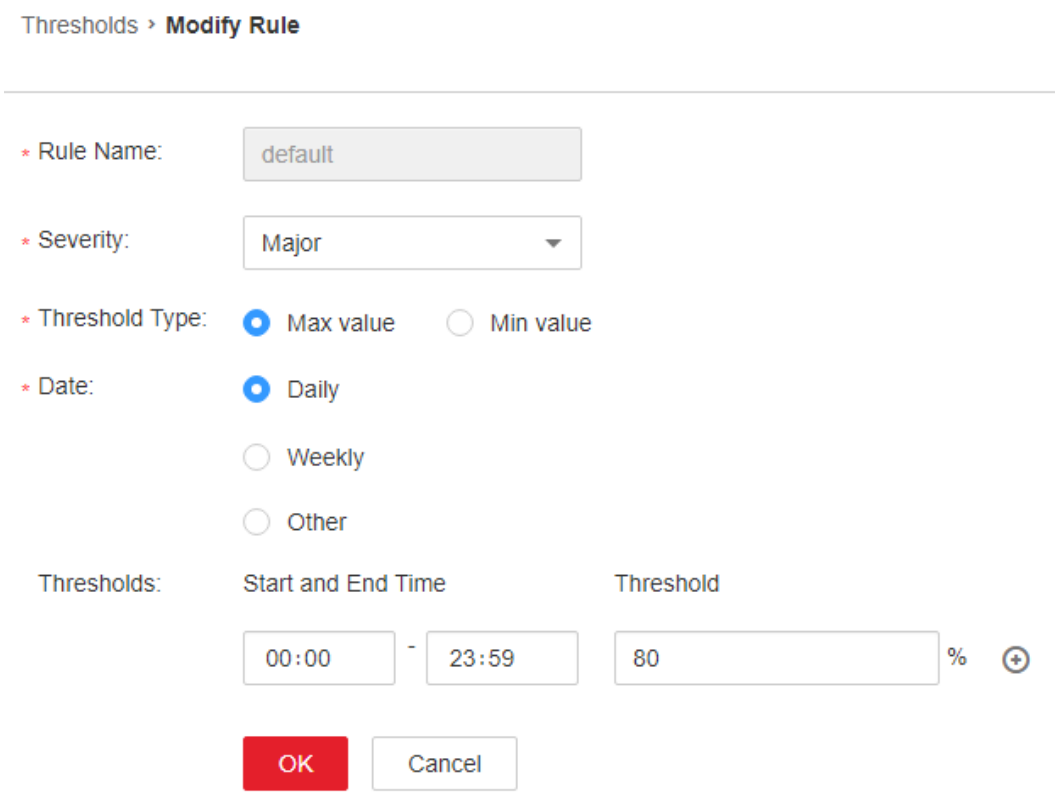
Step 1 On the FusionInsight Manager, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Reading** > **Read Throughput Rate** and check whether the alarm threshold is set properly. (By default, 80% is a proper value. However, users can configure the value as required.)

- If yes, go to **Step 2**.
- If no, go to **Step 4**.

Step 2 Based on actual usage condition, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Reading** > **Read Throughput Rate** and click **Modify** in the **Operation** column to modify the alarm threshold.

For details, see **Figure 10-29**.

Figure 10-29 Setting alarm thresholds



Step 3 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the network port rate can meet the service requirements.

Step 4 On FusionInsight Manager, click ⌄ in the row where the alarm is located in the real-time alarm list and obtain the IP address of the host and the network port name for which the alarm is generated.

Step 5 Log in to the host for which the alarm is generated as user **root**.

Step 6 Run the **ethtool** *network port name* command to check the maximum speed of the current network port.

NOTE

In the VM environment, you cannot run a command to query the network port rate. It is recommended that you contact the system administrator to confirm whether the network port rate meets the requirements.

Step 7 If the network read throughput rate exceeds the threshold, contact the system administrator to increase the network port rate.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log > Download**.

Step 10 Select **OMS** from the **Service** and click **OK**.

Step 11 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 12 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.28 ALM-12050 Network Write Throughput Rate Exceeds the Threshold

Description

The system checks the network write throughput rate every 30 seconds and compares the actual throughput rate with the threshold (the default threshold is 80%). This alarm is generated when the system detects that the network write throughput rate exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Throughput Rate**.

When the **Trigger Count** is 1, this alarm is cleared when the network write throughput rate is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the network write throughput rate is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12050	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NetworkCardName	Specifies the network port for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The service system runs improperly or is unavailable.

Possible Causes

- The alarm threshold is set improperly.
- The network port rate cannot meet the current service requirements.

Procedure

Check whether the threshold is set properly.

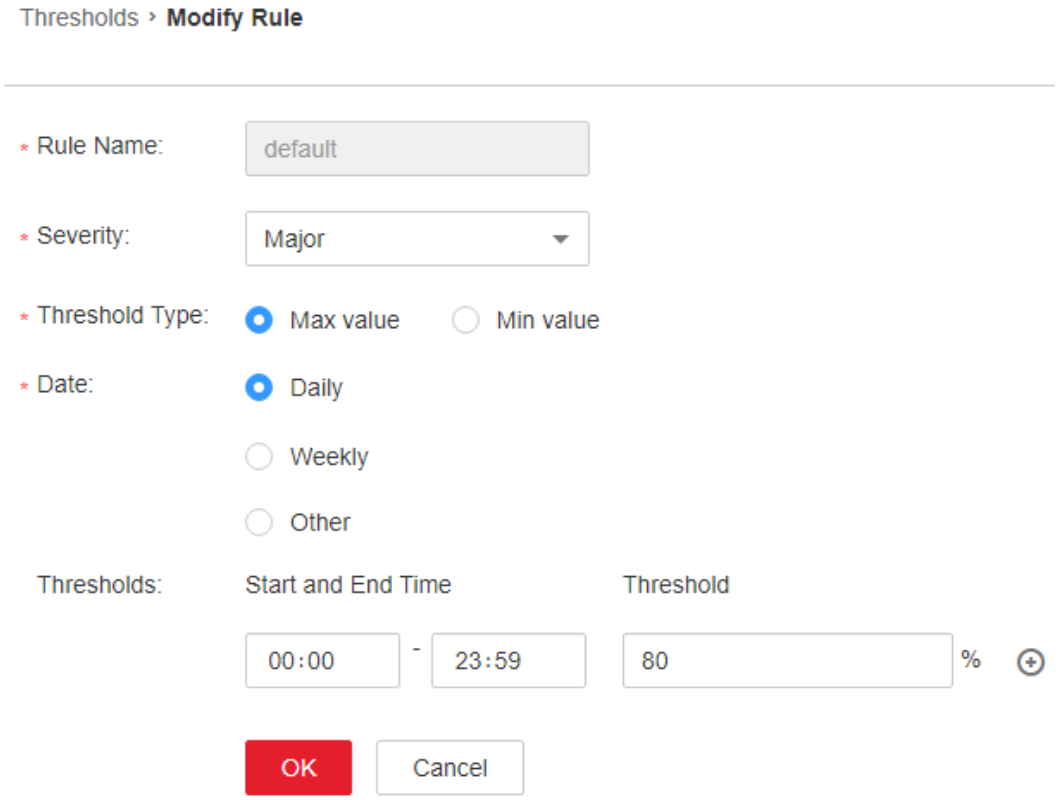
Step 1 On the FusionInsight Manager, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Throughput Rate** and check whether the alarm threshold is set properly. (By default, 80% is a proper value. However, users can configure the value as required.)

- If yes, go to **Step 4**.
- If no, go to **Step 2**.

Step 2 Based on actual usage condition, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Writing** > **Write Throughput Rate** and click **Modify** in the **Operation** column to modify the alarm threshold.

For details, see **Figure 10-30**.

Figure 10-30 Setting alarm thresholds



Step 3 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the network port rate can meet the service requirements.

Step 4 On FusionInsight Manager, click ⌄ in the row where the alarm is located in the real-time alarm list and obtain the IP address of the host and the network port name for which the alarm is generated.

Step 5 Log in to the host for which the alarm is generated as user **root**.

Step 6 Run the **ethtool***network port name* command to check the maximum speed of the current network port.

NOTE

In the VM environment, you cannot run a command to query the network port rate. It is recommended that you contact the system administrator to confirm whether the network port rate meets the requirements.

Step 7 If the network write throughput rate exceeds the threshold, contact the system administrator to increase the network port rate.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log > Download**.

Step 10 Select **OMS** from the **Service** and click **OK**.

Step 11 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 12 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.29 ALM-12051 Disk Inode Usage Exceeds the Threshold

Description

The system checks the disk Inode usage every 30 seconds and compares the actual Inode usage with the threshold (the default threshold is 80%). This alarm is generated when the Inode usage exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Disk** > **Disk Inode Usage**.

When the **Trigger Count** is 1, this alarm is cleared when the disk Inode usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the disk Inode usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12051	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
PartitionName	Specifies the disk partition for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Data cannot be properly written to the file system.

Possible Causes

Massive small files are stored in the disk.

Procedure

Massive small files are stored in the disk.

Step 1 On FusionInsight Manager, choose **O&M > Alarm > Alarms** and click ⌄ in the row where the alarm is located in the real-time alarm list and obtain the IP address of the host and the disk partition for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **df -i | grep -iE "***partition name***|**Filesystem" command to check the current disk Inode usage.

```
# df -i | grep -iE "xvda2|Filesystem"
Filesystem      Inodes  IUsed   IFree IUse% Mounted on
/dev/xvda2      2359296  207420 2151876    9% /
```

Step 4 If the Inode usage exceeds the threshold, manually check small files stored in the disk partition and confirm whether these small files can be deleted.

NOTE

Run the **for i in /*; do echo $i; find $i|wc -l; done** command to query the number of files in a partition. Replace **/*** with the specified partition.

```
# for i in /srv/*; do echo $i; find $i|wc -l; done
/srv/BigData
4284
/srv/ftp
1
/srv/www
13
```

- If yes, run the **rm -rf** *Path of the file or folder* to be deleted command to delete the file or folder and go to **Step 5**.

  NOTE

  Deleting a file or folder is a high-risk operation. Ensure that the file or folder is no longer required before performing this operation.

- If no, expand the capacity. Then, perform **Step 5**.

Step 5 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log > Download**.

Step 7 Select **OMS** from the **Service** and click **OK**.

Step 8 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.30 ALM-12052 TCP Temporary Port Usage Exceeds the Threshold

Description

The system checks the TCP temporary port usage every 30 seconds and compares the actual usage with the threshold (the default threshold is 80%). This alarm is generated when the TCP temporary port usage exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Network Status** > **TCP Ephemeral Port Usage**.

When the **Trigger Count** is 1, this alarm is cleared when the TCP temporary port usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the TCP temporary port usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12052	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Services on the host cannot establish external connections, and therefore they are interrupted.

Possible Causes

- The temporary port cannot meet the current service requirements.
- The system is abnormal.

Procedure

Expand the temporary port number range.

Step 1 On FusionInsight Manager, click ⌄ in the row where the alarm is located in the real-time alarm list and obtain the IP address of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **omm**.

Step 3 Run the **cat /proc/sys/net/ipv4/ip_local_port_range |cut -f 1** command to obtain the value of the start port and run the **cat /proc/sys/net/ipv4/ip_local_port_range |cut -f 2** command to obtain the value of the end port. The total number of temporary ports is the value of the end port minus the value of the start port. If the total number of temporary ports is smaller than 28,232, the

random port range of the OS is narrow. Contact the system administrator to increase the port range.

Step 4 Run the **ss -ant 2>/dev/null | grep -v LISTEN | awk 'NR > 2 {print $4}'|cut -d ':' -f 2 | awk '$1 >"***Value of the start port***" {print $1}' | sort -u | wc -l** command to calculate the number of used temporary ports.

Step 5 The formula for calculating the usage of the temporary ports is: Usage of the temporary ports = (Number of used temporary ports/Total number of temporary ports) x 100%. Check whether the temporary port usage exceeds the threshold.

- If yes, go to **Step 7**.
- If no, go to **Step 6**.

Step 6 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Check whether the system environment is abnormal.

Step 7 Run the following command to import the temporary file and view the frequently used ports in the **port_result.txt file**:

netstat -tnp|sort > $BIGDATA_HOME/tmp/port_result.txt

```
netstat -tnp|sort

Active Internet connections (w/o servers)

Proto Recv Send LocalAddress ForeignAddress State PID/ProgramName tcp   0   0 10-120-85-154:45433
10-120-85-154:9866 CLOSE_WAIT 94237/java
tcp   0   0 10-120-85-154:45434  10-120-85-154:9866 CLOSE_WAIT 94237/java
tcp   0   0 10-120-85-154:45435  10-120-85-154:9866 CLOSE_WAIT 94237/java
...
```

Step 8 Run the following command to view the processes that occupy a large number of ports:

ps -ef |grep *PID*

NOTE

- PID is the processes ID queried in **Step 7**.
- Run the following command to collect information about all processes and check the processes that occupy a large number of ports:

  ps -ef > $BIGDATA_HOME/tmp/ps_result.txt

Step 9 After obtaining the administrator's approval, clear the processes that occupy a large number of ports. Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

Collect fault information.

Step 10 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log** > **Download**.

Step 11 Select **OMS** from the **Service** and click **OK**.

Step 12 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 13 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected log information and files **port_result.txt** and **ps_result.txt**. Then, delete the two residual temporary files from the environment.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.31 ALM-12053 Host File Handle Usage Exceeds the Threshold

Description

The system checks the file handle usage every 30 seconds and compares the actual usage with the threshold (the default threshold is 80%). This alarm is generated when the host file handle usage exceeds the threshold for several times (5 times by default) consecutively.

To change the threshold, choose **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Host** > **Host Status** > **Host File Handle Usage**.

When the **Trigger Count** is 1, this alarm is cleared when the host file handle usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the host file handle usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12053	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The I/O operations, such as opening a file or connecting to network, cannot be performed and programs are abnormal.

Possible Causes

- The application process is abnormal. For example, the opened file or socket is not closed.
- The number of file handles cannot meet the current service requirements.
- The system is abnormal.

Procedure

Check information about files opened in processes.

Step 1 On FusionInsight Manager, click ⌄ in the row where the alarm is located in the real-time alarm list and obtain the IP address of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **lsof -n|awk '{print $2}'|sort|uniq -c|sort -nr|more** command to check the process that occupies excessive file handles.

Step 4 Check whether the processes in which a large number of files are opened are normal. For example, check whether there are files or sockets not closed.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 Release the abnormal processes that occupy too many file handles.

Step 6 Five minutes later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Increase the number of file handles.

Step 7 On FusionInsight Manager, click ⌄ in the row where the alarm is located in the real-time alarm list and obtain the IP address of the host for which the alarm is generated.

Step 8 Log in to the host for which the alarm is generated as user **root**.

Step 9 Contact the system administrator to increase the number of system file handles.

Step 10 Run the **cat /proc/sys/fs/file-nr** command to view the used handles and the maximum number of file handles. The first value is the number of used handles, the third value is the maximum number. Please check whether the usage exceeds the threshold.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

```
# cat /proc/sys/fs/file-nr
12704 0 640000
```

Step 11 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Check whether the system environment is abnormal.

Step 12 Contact the system administrator to check whether the operating system is abnormal.

- If yes, go to **Step 13** to rectify the fault.
- If no, go to **Step 14**.

Step 13 Wait for 5 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Collect fault information.

Step 14 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log** > **Download**.

Step 15 Select **OMS** from the **Service** and click **OK**.

Step 16 Set **Host** to the node for which the alarm is generated and the active OMS node.

Step 17 Click ✎ in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.32 ALM-12054 Invalid Certificate File

Description

The system checks whether the certificate file is invalid (has expired or is not yet valid) on 23:00 every day. This alarm is generated when the certificate file is invalid.

This alarm is cleared when the status of the newly imported certificate is valid.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12054	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Some functions are unavailable.

Possible Causes

No certificate (CA certificate, HA root certificate, HA user certificate, Gaussdb root certificate, or Gaussdb user certificate) is imported to the system, the certificate fails to be imported, or the certificate file is invalid.

Procedure

Check the alarm cause.

Step 1 On FusionInsight Manager, locate the target alarm in the real-time alarm list and click ⌄.

View **Additional Information** to obtain the additional information about the alarm.

- If **CA Certificate** is displayed in the additional alarm information, log in to the active OMS management node as user **omm** and go to **Step 2**.
- If **HA root Certificate** is displayed in the additional information, view **Location** to obtain the name of the host involved in this alarm. Then, log in to the host as user **omm** and go to **Step 3**.
- If **HA server Certificate** is displayed in the additional information, view **Location** to obtain the name of the host involved in this alarm. Then, log in to the host as user **omm** and go to **Step 4**.

Check the validity period of the certificate files in the system.

Step 2 Check whether the current system time is in the validity period of the CA certificate.

Run the **bash ${CONTROLLER_HOME}/security/cert/conf/querycertvalidity.sh** command to check the effective time and due time of the CA root certificate.

- If yes, go to **Step 7**.
- If no, go to **Step 5**.

Step 3 Check whether the current system time is in the validity period of the HA root certificate.

Run the **openssl x509 -noout -text -in ${CONTROLLER_HOME}/security/certHA/root-ca.crt** command to check the effective time and due time of the HA root certificate.

- If yes, go to **Step 7**.
- If no, go to **Step 6**.

Step 4 Check whether the current system time is in the validity period of the HA user certificate.

Run the **openssl x509 -noout -text -in ${CONTROLLER_HOME}/security/certHA/server.crt** command to check the effective time and due time of the HA user certificate.

- If yes, go to **Step 7**.
- If no, go to **Step 6**.

The following is an example of the effective time and due time of a CA or HA certificate:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            97:d5:0e:84:af:ec:34:d8
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=CN, ST=xxx, L=yyy, O=zzz, OU=IT, CN=HADOOP.COM
        Validity
            Not Before: Dec 13 06:38:26 2016 GMT          // Effective time
            Not After : Dec 11 06:38:26 2026 GMT          // Due time
```

Import certificate files.

Step 5 Import a new CA certificate file.

Apply for or generate a new CA certificate file and import it to the system. The alarm is automatically cleared after the CA certificate is imported. Check whether this alarm is reported again during periodic check.

- If yes, go to **Step 7**.
- If no, no further action is required.

Step 6 Import a new HA certificate file.

Apply for or generate a new HA certificate file and import it to the system. The alarm is automatically cleared after the CA certificate is imported. Check whether this alarm is reported again during periodic check.

- If yes, go to .**Step 7**
- If no, no further action is required.

Collect the fault information.

Step 7 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 8 In the **Services** area, select **Controller**, **OmmServer**, **OmmCore**, and **Tomcat**, and click **OK**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.33 ALM-12055 The Certificate File Is About to Expire

Description

The system checks the certificate file on 23:00 every day. This alarm is generated if the certificate file is about to expire with a validity period less than days set in the alarm threshold.

This alarm is reported if the status of the newly imported certificate is valid.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12055	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Some functions are unavailable.

Possible Causes

The remaining validity period of a system certificate (CA certificate, HA root certificate, HA user certificate, Gaussdb root certificate, or Gaussdb user certificate) is smaller than the alarm threshold.

Procedure

Check the alarm cause.

Step 1 On FusionInsight Manager, locate the target alarm in the real-time alarm list and click ⌄.

View **Additional Information** to obtain the additional information about the alarm.

- If **CA Certificate** is displayed in the additional alarm information, log in to the active OMS management node as user **omm** and go to **Step 2**.
- If **HA root Certificate** is displayed in the additional information, view **Location** to obtain the name of the host involved in this alarm. Then, log in to the host as user **omm** and go to **Step 3**.

- If **HA server Certificate** is displayed in the additional information, view **Location** to obtain the name of the host involved in this alarm. Then, log in to the host as user **omm** and go to **Step 4**.

Check the validity period of the certificate files in the system.

Step 2 Check whether the remaining validity period of the CA certificate is smaller than the alarm threshold.

Run the **bash ${CONTROLLER_HOME}/security/cert/conf/querycertvalidity.sh** command to check the effective time and due time of the CA root certificate.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 3 Check whether the remaining validity period of the HA root certificate is smaller than the alarm threshold.

Run the **openssl x509 -noout -text -in ${CONTROLLER_HOME}/security/certHA/root-ca.crt** command to check the effective time and due time of the HA root certificate.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 4 Check whether the remaining validity period of the HA user certificate is smaller than the alarm threshold.

Run the **openssl x509 -noout -text -in ${CONTROLLER_HOME}/security/certHA/server.crt** command to check the effective time and due time of the HA user certificate.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

The following is an example of the effective time and due time of a CA or HA certificate:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            97:d5:0e:84:af:ec:34:d8
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=CN, ST=xxx, L=yyy, O=zzz, OU=IT, CN=HADOOP.COM
        Validity
            Not Before: Dec 13 06:38:26 2016 GMT            // Effective time
            Not After : Dec 11 06:38:26 2026 GMT            // Due time
```

Import certificate files.

Step 5 Import a new CA certificate file.

Apply for or generate a new CA certificate file and import it to the system. Manually clear the alarm and check whether this alarm is generated again during periodic check.

- If yes, go to **Step 7**.
- If no, no further action is required.

Step 6 Import a new HA certificate file.

Apply for or generate a new HA certificate file and import it to the system. Manually clear the alarm and check whether this alarm is generated again during periodic check.

- If yes, go to **Step 7**.
- If no, no further action is required.

Collect the fault information.

Step 7 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 8 In the **Services** area, select **Controller**, **OmmServer**, **OmmCore**, and **Tomcat**, and click **OK**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.34 ALM-12057 Metadata Not Configured with the Task to Periodically Back Up Data to a Third-Party Server

Description

After the system is installed, it checks whether the task for periodically backing up metadata to the third-party server, and then performs the check hourly. If the task for periodically backing up metadata to a third-party server is not configured, a critical alarm is generated.

This alarm is cleared when a user creates such a backup task.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12057	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

If metadata is not backed up to a third-party server, metadata cannot be restored if both the active and standby management nodes of the cluster are faulty and local backup data is lost.

Possible Causes

Metadata is not configured with the task to periodically back up data to a third-party server.

Procedure

Step 1 On the FusionInsight Manager portal choose **O&M > Alarm > Alarms**.

Step 2 In the alarm list, click ⌄ in the row where the alarm is located and identify the data module from which the alarm is generated based on **Additional Information**.

Step 3 Choose **O&M** > **Backup and Restoration > Backup Management** > **Create**.

Step 4 Configure a backup task. The backup data to be configured is consistent with the data in Additional Information of the alarm.

Step 5 After the backup task is created successfully, wait for two minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information

Step 6 On FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 7 In the **Service** area, select **Controller** and click **OK**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.35 ALM-12061 Process Usage Exceeds the Threshold

Description

The system checks the usage of the omm process every 30 seconds. Users can run the **ps -o nlwp, pid, args, -u omm | awk '{sum+=$1} END {print "", sum}'** command to obtain the number of concurrent processes of user **omm**. Run the **ulimit -u**command to obtain the maximum number of processes that can be simultaneously opened by user **omm**. Divide the number of concurrent processes by the maximum number to obtain the process usage of user **omm**. The process usage has a default threshold. This alarm is generated when the process usage exceeds the threshold.

If **Trigger Count** is **3** and the process usage is less than or equal to the threshold, this alarm is cleared. If **Trigger Count** is greater than **1**and the process usage is less than or equal to 90% of the threshold, this alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12061	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

- Switch to user **omm** fails.
- New omm process cannot be created.

Possible Causes

- The alarm threshold is improperly configured.
- The maximum number of processes (including threads) that can be concurrently opened by user **omm** is inappropriate.
- An excessive number of threads are opened at the same time.

Procedure

Check whether the alarm threshold or alarm hit number is properly configured.

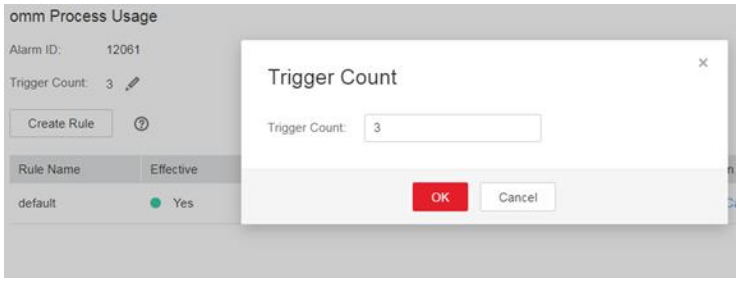
Step 1 On the FusionInsight Manager, change the alarm threshold and **Trigger Count** based on the actual CPU usage.

Specifically, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Host**> **Process** > **omm Process Usage** to change Trigger Count, as shown in **Figure 10-31**.

NOTE

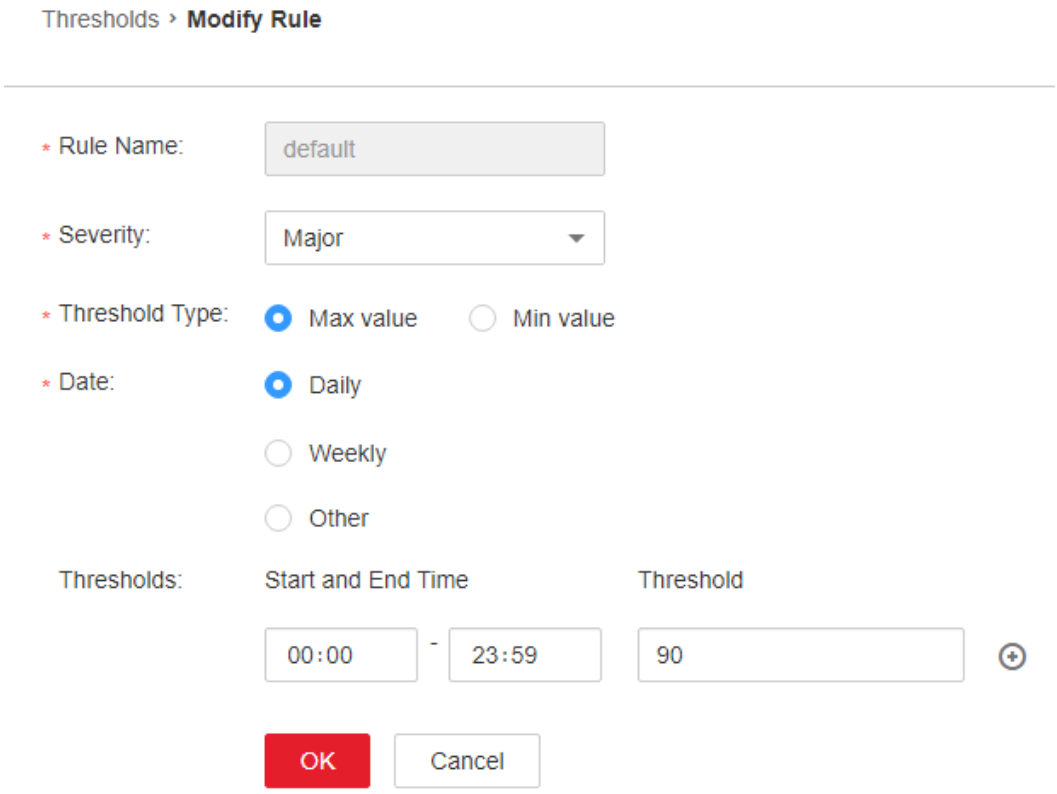
The alarm is generated when the process usage exceeds the threshold for the times specified by **Trigger Count**.

Figure 10-31 Setting Trigger Count



Set the alarm threshold based on the actual process usage. To check the process usage, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Host**> **Process** > **omm Process Usage**, as shown in **Figure 10-32**.

Figure 10-32 Setting an alarm threshold



Step 2 2 minutes later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 3**.

Check whether the maximum number of processes (including threads) opened by user omm is appropriate.

Step 3 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host for which the alarm is generated.

Step 4 Log in to the host where the alarm is generated as user **root**.

Step 5 Run the **su - omm** command to switch to user **omm**.

Step 6 Run the **ulimit -u** command to obtain the maximum number of threads that can be concurrently opened by user **omm** and check whether the number is greater than or equal to 60000.

- If it is, go to **Step 8**.
- If it is not, go to **Step 7**.

Step 7 Run the **ulimit -u 60000** command to change the maximum number to 60000. Two minutes later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 12**.

Check whether an excessive number of processes are opened at the same time.

Step 8 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host for which the alarm is generated.

Step 9 Log in to the host where the alarm is generated as user **root**.

Step 10 Run the **ps -o nlwp, pid, lwp, args, -u omm|sort -n** command to check the numbers of threads used by the system. The result is sorted based on the thread number. Analyze the top 5 thread numbers and check whether the threads are incorrectly used. If they are, contact maintenance personnel to rectify the fault. If they are not, run the **ulimit -u** command to change the maximum number to be greater than 60000.

Step 11 Five minutes later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 12**.

Collect fault information.

Step 12 On the FusionInsight Manager home page of the active clusters, choose **O&M** > **Log** > **Download**.

Step 13 Select **OmmServer** and **NodeAgent** from the **Service** and click **OK**.

Step 14 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 15 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.36 ALM-12062 OMS Parameter Configurations Mismatch with the Cluster Scale

Description

The system checks whether the OMS parameter configurations match with the cluster scale at each top hour. If the OMS parameter configurations do not meet the cluster scale requirements, the system generates this alarm. This alarm is automatically cleared when the OMS parameter configurations are modified.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12062	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the name of the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The OMS configuration is not modified when the cluster is installed or the system capacity is expanded.

Possible Causes

The OMS parameter configurations mismatch with the cluster scale.

Procedure

Check whether the OMS parameter configurations match with the cluster scale.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host for which the alarm is generated.

Step 2 Log in to the host where the alarm is generated as user **root**.

Step 3 Run the **su - omm** command to switch to user **omm**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/controller/scriptlog/modify_manager_param.log** command to open the log file and search for the log file containing the following information: Current oms configurations cannot support *xx* nodes. In the information, *xx* indicates the number of nodes in the cluster.

Step 5 Optimize the current cluster configuration by following the instructions in **Optimizing Manager Configurations Based on the Number of Cluster Nodes**.

Step 6 One hour later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Select **Controller** from the **Service** and click **OK**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

Optimizing Manager Configurations Based on the Number of Cluster Nodes

Step 1 Log in to the active Manager node as user **omm**.

Step 2 Run the following command to switch the directory:

cd ${BIGDATA_HOME}/om-server/om/sbin

Step 3 Run the following command to view the current Manager configurations.

sh oms_config_info.sh -q

Step 4 Run the following command to specify the number of nodes in the current cluster.

Command format: **sh oms_config_info.sh -s** *number of nodes*

Example:

sh oms_config_info.sh -s 1000

Enter **y** as prompted.

```
The following configurations will be modified:
    Module      Parameter         Current         Target
    Controller  controller.Xmx    4096m       =>  16384m
    Controller  controller.Xms    1024m       =>  8192m
    Controller  controller.node.heartbeat.error.threshold     30000               =>  60000
    Pms         pms.mem           8192m       =>  10240m
Do you really want to do this operation? (y/n):
```

The configurations are updated successfully if the following information is displayed:

```
...
Operation has been completed. Now restarting OMS server.                [done]
Restarted oms server successfully.
```

NOTE

- OMS is automatically restarted during the configuration update process.
- Clusters with similar quantities of nodes have same Manager configurations. For example, when the number of nodes is changed from 100 to 101, no configuration item needs to be updated.

----End

10.13.37 ALM-12063 Unavailable Disk

Description

The system checks whether the data disk of the current host is available at the top of each hour. The system creates files, writes files, and deletes files in the mount directory of the disk. If the operations fail, the alarm is generated. If the operations succeed, the disk is available, and the alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12063	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the name of the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
DiskName	Specifies the disk for which the alarm is generated.

Impact on the System

Data read or write on the data disk fails, and services are abnormal.

Possible Causes

- The permission of the disk mount directory is abnormal.
- There are disk bad sectors.

Procedure

Check whether the permission of the disk mount directory is normal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host and **DiskName** for the disk for which the alarm is generated.

Step 2 Log in to the host where the alarm is generated as user **root**.

Step 3 Run the **df -h |grep DiskName** command to obtain the mount point and check whether the permission of the mount directory is unwritable or unreadable.

- If it is, go to **Step 4**.
- If it is not, go to **Step 8**.

NOTE

If the permission of the mount directory is 000 or the owner is **root**, the mount directory is unreadable and unwritable.

Step 4 Modify the directory permission.

Step 5 One hour later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Step 6 Contact hardware engineers to rectify the disk.

Step 7 One hour later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 8**.

Collect fault information.

Step 8 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 9 Select **NodeAgent** from the **Service** and click **OK**.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.38 ALM-12064 Host Random Port Range Conflicts with Cluster Used Port

Alarm Description

The system checks whether the random port range of the host conflicts with the range of ports used by the Cluster system every hour. The alarm is generated if they conflict. The alarm is automatically cleared when the random port range of the host is changed to the normal range.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12064	Major	Yes

Parameters

Parameter	Description
Source	Specifiestheclusterorsystemforwhich-thealarmisgenerated.
ServiceName	Specifies the name of the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The default port of the Cluster system is occupied. As a result, some processes fail to be started.

Possible Causes

The random port range configuration is modified.

Procedure

Check the random port range of the system.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host for which the alarm is generated.

Step 2 Log in to the host where the alarm is generated as user **root**.

Step 3 Run the **cat /proc/sys/net/ipv4/ip_local_port_range** command to obtain the random port range of the host and check whether the minimum value is smaller than 32768.

- If it is, go to **Step 4**.
- If it is not, goto **Step 7**.

Step 4 Run the **vim /etc/sysctl.conf** command to change the value of **net.ipv4.ip_local_port_range** to **32768 61000**. If this parameter does not exist, add the following configuration: **net.ipv4.ip_local_port_range = 32768 61000**.

Step 5 Run the **sysctl -p /etc/sysctl.conf** command for the modification to take effect.

Step 6 One hour later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Select **NodeAgent** for **Service** and click **OK**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.39 ALM-12066 Trust Relationships Between Nodes Become Invalid

Description

The system checks whether the trust relationship between the active OMS node and other Agent nodes is normal every hour. The alarm is generated if the mutual trust fails. This alarm is automatically cleared if this problem is resolved.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12066	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Some operations on the management plane may be abnormal.

Possible Causes

- The **/etc/ssh/sshd_config** configuration file is damaged.
- The password of user **omm** has expired.

Procedure

Check the status of the /etc/ssh/sshd_config configuration file.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm and click ⌄ to view the host list in the alarm details.

Step 2 Log in to the active OMS node as user **omm**.

Step 3 Run the **ssh** command, for example, **ssh *host2***, on each node in the alarm details to check whether the connection fails. (***host2*** is a node other than the OMS node in the alarm details.)

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 Open the **/etc/ssh/sshd_config** configuration file on host2 and check whether **AllowUsers** or **DenyUsers** is configured for other nodes.

- If yes, go to **Step 5**.
- If no, contact OS experts.

Step 5 Modify the whitelist or blacklist to ensure that user **omm** is in the whitelist or not in the blacklist. Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check the status of the password of user omm.

Step 6 Check the interaction information of the **ssh** command.

- If the password of user **omm** is required, go to **Step 7**.
- If message "Enter passphrase for key '/home/omm/.ssh/id_rsa':" is displayed, go to **Step 9**.

Step 7 Check the trust list (**/home/omm/.ssh/authorized_keys**) of user **omm** on the OMS node and host2 node. Check whether the trust list contains the public key file (**/home/omm/.ssh/id_rsa.pub**) of user **omm** on the peer host.

- If yes, contact OS experts.
- If no, add the public key of user **omm** of the peer host to the trust list of the local host.

Step 8 Add the public key of user **omm** of the peer host to the trust list of the local host. Run the **ssh** command, for example, **ssh host2**, on each node in the alarm details to check whether the connection fails. (***host2*** is a node other than the OMS node in the alarm details.)

- If yes, go to **Step 9**.
- If no, check whether the alarm is cleared. If the alarm is cleared, no further action is required; otherwise, go to **Step 9**.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Select **Controller** for **Service** and click **OK**.

Step 11 Click  in the upper right corner to set the log collection time range. Generally, the time range is 10 minutes before and after the alarm generation time. Click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

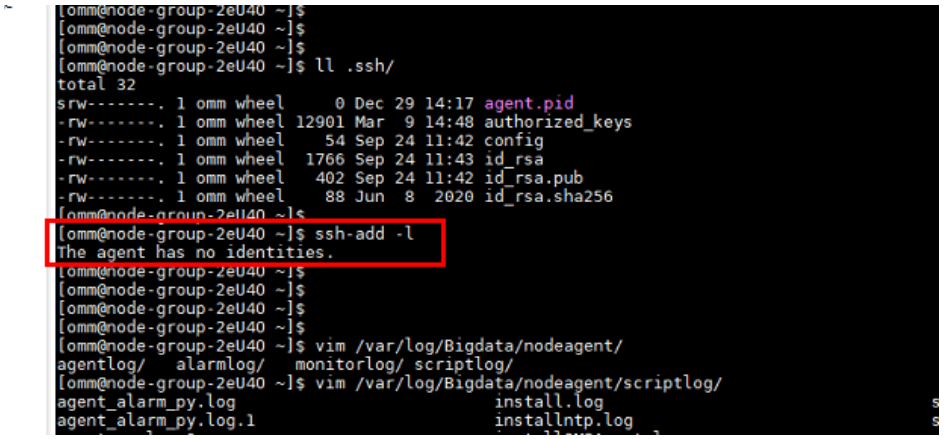
Related Information

Perform the following steps to handle abnormal trust relationships between nodes:

NOTICE

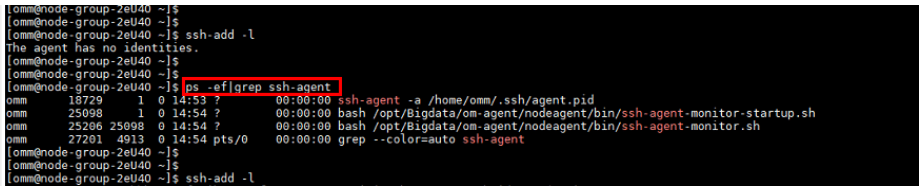
- Perform this operation as user **omm**.
- If the network between nodes is disconnected, rectify the network fault first. Check whether the two nodes are connected to the same security group and whether **hosts.deny** and **hosts.allow** are set.

1. Run the **ssh-add -l** command on both nodes to check whether any identities exist.

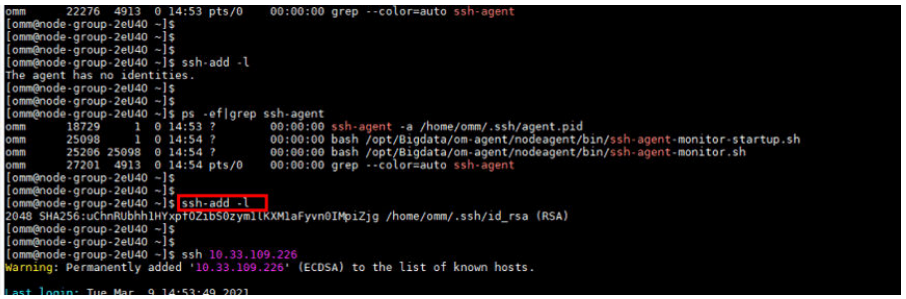


  - If yes, go to **4**.
  - If no, go to **2**.

2. If no identities are displayed, run the **ps -ef|grep ssh-agent** command to find the **ssh-agent** process, stop the process, and wait for the process to automatically restart.



3. Run the **ssh-add -l** command to check whether the identities have been added. If yes, manually run the **ssh** command to check whether the trust relationship is normal.



4. If identities exist, check whether the **/home/omm/.ssh/authorized_keys** file contains the information in the **/home/omm/.ssh/id_rsa.pub** file of the peer node. If it does not, manually add the information.
5. Check whether the permissions on the files in the **/home/omm/.ssh** directory are modified.
6. Check the **/var/log/Bigdata/nodeagent/scriptlog/ssh-agent-monitor.log** file.
7. If the **/home** directory of user **omm** is deleted, contact MRS support personnel for assistance.

10.13.40 ALM-12067 Tomcat Resource Is Abnormal

Alarm Description

HA checks the Tomcat resources of Manager every 85 seconds. This alarm is generated when HA detects that the Tomcat resources are abnormal for 2 consecutive times.

This alarm is cleared when the Tomcat resource is normal.

Resource Type of Tomcat is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new Tomcat resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12067	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the name of the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The Tomcat process repeatedly restarts.

Possible Causes

- The Tomcat directory permission is incorrect. The Tomcat process is abnormal.

Procedure

Check whether the Tomcat directory permission is normal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host for which the alarm is generated.

Step 2 Log in to the host where the alarm is generated as user **root**.

Step 3 Run the **su - omm** command to switch to user **omm**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/ha/scriptlog/tomcat.log** command to check whether the Tomcat resource log contains the keyword "Cannot find *XXX*" and rectify the file permission based on the keyword.

Step 5 5 minutes later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Select **OmmServer** and **Tomcat** for **Service** and click **OK**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.41 ALM-12068 ACS Resource Is Abnormal

Alarm Description

HA checks the acs resources of Manager every 80 seconds. This alarm is generated when HA detects that the acs resources are abnormal for two consecutive times.

This alarm is cleared when the ACS resource is normal.

Resource Type of ACS is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new ACS resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12068	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the name of the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The ACS process repeatedly restarts, which may cause the FusionInsight Manager login failure.

Possible Causes

The ACS process is abnormal.

Procedure

Check whether the ACS process is abnormal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su -omm** command and then the **sh ${BIGDATA_HOME} /om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the ACS resources managed by the HA is normal. In the single-node system, the ACS resource is in the normal state. In the dual-node system, the ACS resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 6**.
- If it is not, go to **Step 4**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/ha/scriptlog/acs.log** command to view the ACS resource logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 5 One hour later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.42 ALM-12069 AOS Resource Is Abnormal

Alarm Description

HA checks the aos resources of Manager every 81 seconds. This alarm is generated when HA detects that the aos resources are abnormal for 2 consecutive times.

This alarm is cleared when the AOS resource is normal.

Resource Type of AOS is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new AOS resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12069	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the name of the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The AOS process repeatedly restarts, which may cause the FusionInsight Manager login failure.

Possible Causes

The AOS process is abnormal.

Procedure

Check the random port range of the system.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su -omm** command and then the **sh ${BIGDATA_HOME} /om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the AOS resources managed by the HA is normal. In the single-node system, the AOS resource is in the normal state. In the dual-node system, the AOS resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 6**.
- If it is not, go to **Step 4**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/ha/scriptlog/aos.log** command to view the AOS resource logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 5 Five minutes later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour before and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.43 ALM-12070 Controller Resource Is Abnormal

Alarm Description

HA checks the controller resources of Manager every 80 seconds. This alarm is generated when HA detects that the controller resources are abnormal for 2 consecutive times.

This alarm is cleared when the Controller resource is normal.

Resource Type of Controller is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new Controller resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12070	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the name of the service for which the alarm is generated.

Parameter	Description
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The Controller process repeatedly restarts, which may cause the FusionInsight Manager login failure.

Possible Causes

The Controller process is abnormal.

Procedure

Check whether the controller process is normal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su - omm** command to switch to user **omm**.Run the **sh ${BIGDATA_HOME}/om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the Controller resources managed by the HA is normal. In the single-node system, the Controller resource is in the normal state. In the dual-node system, the Controller resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 6**.
- If it is not, go to **Step 4**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/ha/scriptlog/controller.log** command to view the Controller resource logs, and run the **vi $BIGDATA_LOG_HOME/controller/controller.log** command to view the Controller running logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 5 Five minutes later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Select **Controller** and **OmmServe** for **Service** and click **OK**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour before and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.44 ALM-12071 Httpd Resource Is Abnormal

Description

HA checks the httpd resources of Manager every 120 seconds. This alarm is generated when HA detects that the httpd resources are abnormal for 10 consecutive times.

This alarm is cleared when the httpd resource is normal.

Resource Type of httpd is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new httpd resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12071	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The httpd process is repeatedly restarts, which may lead to the failure to visit the native service UI.

Possible Causes

The httpd process is abnormal.

Procedure

Check whether the httpd process is abnormal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su - omm** command to switch to user **omm**.

Step 4 Run the **sh ${BIGDATA_HOME}/om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the httpd resources managed by the HA is normal. In the single-node system, the httpd resource is in the normal state. In the dual-node system, the httpd resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 7**.
- If it is not, go to **Step 5**.

Step 5 Run the **vi $BIGDATA_LOG_HOME/omm/oms/ha/scriptlog/httpd.log** command to view the httpd resource logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 6 Five minutes later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 9 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 1 hour before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.45 ALM-12072 FloatIP Resource Is Abnormal

Description

HA checks the floatip resources of Manager every 9 seconds. This alarm is generated when HA detects that the floatip resources are abnormal for 3 consecutive times.

This alarm is cleared when the FloatIP resource is normal.

Resource Type of FloatIP is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new FloatIP resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12072	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The FloatIP process is repeatedly restarts, which may lead to the failure to visit the native service UI.

Possible Causes

- The floating IP address is abnormal.

Procedure

Check the floating IP address status of the active management node.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the address of the host for which the alarm is generated and the resource name.

Step 2 Log in to the active management node as user **root**.

Step 3 Run the following command, go to the **${BIGDATA_HOME}/om-server/om/sbin/** directory.

su - omm

cd ${BIGDATA_HOME}/om-server/om/sbin/

Step 4 Run the **sh status-oms.sh** command, and execute the **status-oms.sh** script to check whether the floating IP address of the active FusionInsight Manager is normal. View the command output, locate the row where **ResName** is **floatip**, and check whether the following information is displayed.

For example:

```
10-10-10-160 floatip Normal Normal Single_active
```

- If it is, go to **Step 8**.
- If it is not, go to **Step 5**.

Step 5 Run the **ifconfig** command to check whether the NIC with the floating IP address exists.

- If it does, go to **Step 8**.
- If it does not, go to **Step 6**.

Step 6 Run the **ifconfig** *NIC name Floating IPaddress* netmask *Subnet mask* command to reconfigure the NIC with the floating IP address. (For example, **ifconfig eth0 10.10.10.102 netmask 255.255.255.0**).

Step 7 Five minutes later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 8**.

Collect fault information.

Step 8 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 9 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 10 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 1 hour before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.46 ALM-12073 CEP Resource Is Abnormal

Description

HA checks the cep resources of Manager every 60 seconds. This alarm is generated when HA detects that the cep resources are abnormal for 2 consecutive times.

This alarm is cleared when the CEP resource is normal.

Resource Type of CEP is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new CEP resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12073	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The CEP process repeatedly restarts, causing monitoring data to be abnormal.

Possible Causes

The CEP process is abnormal.

Procedure

Check whether the CEP process is abnormal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su -omm** command and then the **sh ${BIGDATA_HOME}/om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the CEP resources managed by the HA is normal. In the single-node system, the CEP resource is in the normal state. In the dual-node system, the CEP resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 6**.
- If it is not, go to **Step 4**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/cep/cep.log** and **vi $BIGDATA_LOG_HOME/omm/oms/cep/scriptlog/cep_ha.log** commands to view the CEP resource logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 5 Five minutes later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 8 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 1 hour before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.47 ALM-12074 FMS Resource Is Abnormal

Description

HA checks the fms resources of Manager every 60 seconds. This alarm is generated when HA detects that the fms resources are abnormal for 2 consecutive times.

This alarm is cleared when the FMS resource is normal.

Resource Type of FMS is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new FMS resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12074	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The FMS process repeatedly restarts. As a result, alarm information may fail to be reported.

Possible Causes

The FMS process is abnormal.

Procedure

Check whether the FMS process is abnormal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su -omm** command and then the **sh ${BIGDATA_HOME}/om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the FMS resources managed by the HA is normal. In the single-node system, the FMS resource is in the normal state. In the dual-node system, the FMS resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 6**.
- If it is not, go to **Step 4**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/fms/fms.log** and **vi $BIGDATA_LOG_HOME/omm/oms/fms/scriptlog/fms_ha.log** commands to view the FMS resource logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 5 5 minutes later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M**> **Log** > **Download**.

Step 7 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 8 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 1 hour before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.48 ALM-12075 PMS Resource Is Abnormal

Description

HA checks the pms resources of Manager every 55 seconds. This alarm is generated when HA detects that the pms resources are abnormal for three consecutive times.

This alarm is cleared when the PMS resource is normal.

Resource Type of PMS is **Single-active**. Active/standby will be triggered upon resource exceptions. When this alarm is generated, the active/standby switchover is complete and new PMS resources have been enabled on the new active FusionInsight Manager. In this case, this alarm is cleared. This alarm is used to notify users of the cause of the active/standby switchover.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12075	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

- The active/standby FusionInsight Manager switchover occurs.
- The PMS process repeatedly restarts, causing monitoring information to be abnormal.

Possible Causes

The PMS process is abnormal.

Procedure

Check whether the PMS process is abnormal.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the name of the host for which the alarm is generated.

Step 2 Log in to the host for which the alarm is generated as user **root**.

Step 3 Run the **su -omm** command and then the **sh ${BIGDATA_HOME}/om-server/OMS/workspace0/ha/module/hacom/script/status_ha.sh** command to check whether the status of the PMS resources managed by the HA is normal. In the single-node system, the PMS resource is in the normal state. In the dual-node system, the PMS resource is in the normal state on the active node and in the stopped state on the standby node.

- If it is, go to **Step 6**.
- If it is not, go to **Step 4**.

Step 4 Run the **vi $BIGDATA_LOG_HOME/omm/oms/pms/pms.log** and **vi $BIGDATA_LOG_HOME/omm/oms/pms/scriptlog/pms_ha.log** commands to view the PMS resource logs, check whether the keyword **ERROR** exists. Analyze the logs to locate and rectify the fault.

Step 5 Five minutes later, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M**> **Log** > **Download**.

Step 7 Select **Controller** and **OmmServer** for **Service** and click **OK**.

Step 8 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 1 hour before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.49 ALM-12076 GaussDB Resource Is Abnormal

Description

HA checks the Manager database every 10 seconds. This alarm is generated when HA detects that the database is abnormal for 3 consecutive times.

This alarm is cleared when the database is normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12076	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

If databases are abnormal, all core services and related service processes, such as alarms and monitoring functions, are affected.

Possible Causes

An exception occurs in the database.

Procedure

Check the database status of the active and standby management nodes.

Step 1 Log in to the active and standby management nodes respectively as user **root**. Run the **su - ommdba** command to switch to user **ommdba**, and then run the **gs_ctl query** command to check whether the following information is displayed in the command output.

Command output of the active management node:

```
Ha state:
    LOCAL_ROLE: Primary
    STATIC_CONNECTIONS          : 1
    DB_STATE                    : Normal
    DETAIL_INFORMATION          : user/password invalid
Senders info:
    No information
Receiver info:
    No information
```

Command output of the standby management node:

```
Ha state:
    LOCAL_ROLE: Standby
```

```
    STATIC_CONNECTIONS          : 1
    DB_STATE                    : Normal
    DETAIL_INFORMATION          : user/password invalid
Senders info:
    No information
Receiver info:
    No information
```

- If it is, go to **Step 3**.
- If it is not, go to **Step 2**.

Step 2 Contact the network administrator to check whether the network is faulty.

- If it is, go to **Step 3**.
- If it is not, go to **Step 5**.

Step 3 Five minutes later, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Step 4 Log in to the active and standby management nodes, run the **su -omm** command to switch to user **omm**, go to the **${BIGDATA_HOME} /om-server/om/sbin/** directory, and run the **status-oms.sh** script to check whether the floating IP addresses and GaussDB resources of the active and standby FusionInsight Managers are in the status shown in the following figure.

```
acs                 Normal              Normal          Single_active
aos                 Normal              Normal          Single_active
cep                 Normal              Normal          Single_active
controller          Normal              Normal          Single_active
feed_watchdog       Normal              Normal          Double_active
floatip             Normal              Normal          Single_active
fms                 Normal              Normal          Single_active
gaussDB             Active_normal       Normal          Active_standby
heartBeatCheck      Normal              Normal          Single_active
httpd               Normal              Normal          Single_active
iam                 Normal              Normal          Single_active
ntp                 Active_normal       Normal          Active_standby
okerberos           Normal              Normal          Double_active
oldap               Active_normal       Normal          Active_standby
pms                 Normal              Normal          Single_active
tomcat              Normal              Normal          Single_active
acs                 Stopped             Normal          Single_active
aos                 Stopped             Normal          Single_active
cep                 Stopped             Normal          Single_active
controller          Stopped             Normal          Single_active
feed_watchdog       Normal              Normal          Double_active
floatip             Stopped             Normal          Single_active
fms                 Stopped             Normal          Single_active
gaussDB             Standby_normal      Normal          Active_standby
heartBeatCheck      Stopped             Normal          Single_active
httpd               Stopped             Normal          Single_active
```

- If they are, find the alarm in the alarm list and manually clear the alarm.
- If they are not, go to **Step 5**.

Collect fault information.

Step 5 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 6 Select **OmmServer** for **Service** and click **OK**.

Step 7 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 8 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.50 ALM-12077 User omm Expired

Description

The system starts at 00:00 every day to check whether user **omm** has expired every eight hours. This alarm is generated if the user account has expired.

This alarm is cleared when the expiration time of user **omm** is changed and the user account status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12077	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

User **omm** has expired. The node trust relationship is unavailable, and FusionInsight Manager cannot manage the services.

Possible Causes

User **omm** has expired.

Procedure

Check whether user omm in the system has expired.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l omm**command to view the information about the password of user **omm**.

Step 2 View the value of **Account expires** to check whether the user configurations have expired.

NOTE

If the parameter value is **never**, the user configurations never expire.

- If they do, go to **Step 3**.
- If they do not, go to **Step 4**.

Step 3 Run the **chage -E 'yyyy-MM-dd' omm** command to set the expiration time of user **omm**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.51 ALM-12078 Password of User omm Expired

Description

The system starts at 00:00 every day to check whether the password of user **omm** has expired every 8 hours. This alarm is generated if the password has expired.

This alarm is cleared when the expiration time of user **omm** password is changed and the user password status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12078	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The password of user **omm** has expired. The node trust relationship is unavailable, and FusionInsight Manager cannot manage the services.

Possible Causes

The password of user **omm** has expired.

Procedure

Check whether the password of user omm in the system has expired.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l omm**command to view the information about the password of user **omm**.

Step 2 View the value of **Password expires** to check whether the user configurations have expired.

NOTE

If the parameter value is **never**, the user configurations never expire.

- If they do, go to **Step 3**.
- If they do not, go to **Step 4**.

Step 3 Run the **chage -M '*days*' omm** command to set the validity period of the password for user **omm**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M**> **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.52 ALM-12079 User omm Is About to Expire

Description

The system starts at 00:00 every day to check whether user **omm** is about to expire every 8 hours. This alarm is generated if the user account will expire no less than 15 days later.

This alarm is cleared when the expiration time of user **omm** is changed and the user account status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12079	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

User **omm** has expired. The node trust relationship is unavailable, and FusionInsight Manager cannot manage the services.

Possible Causes

The account of user **omm** is about to expire.

Procedure

Check whether user omm is about to expire.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l omm** command to view the information about the password of user **omm**.

Step 2 View the value of **Account expires** to check whether the user configurations are about to expire.

NOTE

If the parameter value is **never**, the user and password are valid permanently; if the value is a date, check whether the user and password are about to expire within 15 days.

- If they are, go to **Step 3**.
- If they are not, go to **Step 4**.

Step 3 Run the **chage -E** *'yyyy-MM-dd'* **omm** command to set the validity period of user **omm**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.53 ALM-12080 Password of User omm Is About to Expire

Description

The system starts at 00:00 every day to check whether the password of user **omm** is about to expire every 8 hours. This alarm is generated if the password will expire no less than 15 days later.

This alarm is cleared when the expiration time of user **omm** password is reset and the user password status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12080	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The password of user **omm** has expired. The node trust relationship is unavailable, and FusionInsight Manager cannot manage the services.

Possible Causes

The password of user **omm** is about to expire.

Procedure

Check whether the password of user omm in the system is about to expire.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l omm**command to view the information about the password of user **omm**.

Step 2 View the value of **Password expires** to check whether the user configurations are about to expire.

NOTE

If the parameter value is **never**, the user and password are valid permanently; if the value is a date, check whether the user and password are about to expire within 15 days.

- If they are, go to **Step 3**.
- If they are not, go to **Step 4**.

Step 3 Run the **chage -M '***days***' omm** command to set the validity period of the password for user **omm**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M**> **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.54 ALM-12081 User ommdba Expired

Description

The system starts at 00:00 every day to check whether user **ommdba** has expired every 8 hours. This alarm is generated if the user account has expired.

This alarm is cleared when the expiration time of user **ommdba** is reset and the user account status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12081	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The OMS database cannot be managed and data cannot be accessed.

Possible Causes

The account of user **ommdba** for the host has expired.

Procedure

Check whether user ommdba has expired.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l ommdba** command to view the information about the password of user **ommdba**.

Step 2 View the value of **Account expires** to check whether the user configurations have expired.

NOTE

If the parameter value is **never**, the user and password are valid permanently; if the value is a date, check whether the user and password have expired.

- If they do, go to **Step 3**.
- If they do not, go to **Step 4**.

Step 3 Run the **chage -E** *'yyyy-MM-dd'* **omm** command to set the validity period of user **ommdba**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.55 ALM-12082 User ommdba Is About to Expire

Description

The system starts at 00:00 every day to check whether user **ommdba** is about to expire every 8 hours. This alarm is generated if the user account will expire no less than 15 days later.

This alarm is cleared when the expiration time of user **ommdba** is reset and the user account status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12082	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The OMS database cannot be managed and data cannot be accessed.

Possible Causes

The account of user **ommdba** for the host is about to expire.

Procedure

Check whether user ommdba is about to expire.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l ommdba** command to view the information about user **ommdba**.

Step 2 View the value of **Account expires** to check whether the user configurations are about to expire.

NOTE

If the parameter value is **never**, the user and password are valid permanently; if the value is a date, check whether the user and password are about to expire within 15 days.

- If they are, go to **Step 3**.
- If they are not, go to **Step 4**.

Step 3 Run the **chage -E** *'yyyy-MM-dd'* **ommdba** command to set the validity period of user **ommdba**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.56 ALM-12083 Password of User ommdba Is About to Expire

Description

The system starts at 00:00 every day to check whether the password of user **ommdba** is about to expire every 8 hours. This alarm is generated if the password is about to expire no less than 15 days later.

This alarm is cleared when the expiration time of user **ommdba** password is reset and the user password status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12083	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The OMS database cannot be managed and data cannot be accessed.

Possible Causes

The password of user **ommdba** is about to expire.

Procedure

Check whether the password of user ommdba in the system is about to expire.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l ommdba** command to view the information about the password of user **ommdba**.

Step 2 View the value of **Password expires** to check whether the user configurations are about to expire.

NOTE

If the parameter value is **never**, the user and password are valid permanently; if the value is a date, check whether the user and password are about to expire within 15 days.

- If they are, go to **Step 3**.
- If they are not, go to **Step 4**.

Step 3 Run the **chage -M** *'days'* **ommdba** command to set the validity period of the password for user **ommdba**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.57 ALM-12084 Password of User ommdba Expired

Description

The system starts at 00:00 every day to check whether the password of user **ommdba** has expired every 8 hours. This alarm is generated if the password has expired.

This alarm is cleared when the expiration time of user **ommdba** password is reset and the user password status becomes normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12084	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The password of user **ommdba** has expired. The node trust relationship is unavailable, and FusionInsight Manager cannot manage the services.

Possible Causes

The password of user **ommdba** for the host has expired.

Procedure

Check whether the password of user ommdba in the system has expired.

Step 1 Log in to the faulty node as user **root**.

Run the **chage -l ommdba** command to view the information about the password of user **ommdba**.

Step 2 View the value of **Password expires** to check whether the user configurations have expired.

NOTE

If the parameter value is **never**, the user and password are valid permanently; if the value is a date, check whether the user and password have expired.

- If they do, go to **Step 3**.
- If they do not, go to **Step 4**.

Step 3 Run the **chage -M** *'days'* **ommdba** command to set the validity period of the password for user **ommdba**. Eight hours later, check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select **NodeAgent** for **Service** and click **OK**.

Step 6 Click in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.58 ALM-12085 Service Audit Log Dump Failure

Description

The system dumps service audit logs at 03:00 every day and stores them on the OMS node. This alarm is generated when the dump fails. This alarm is cleared when the next dump succeeds.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12085	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The service audit logs may be lost.

Possible Causes

- The service audit logs are oversized.
- The OMS backup storage space is insufficient.
- The storage space of a host where the service is located is insufficient.

Procedure

Check whether the service audit logs are oversized.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and view the IP address of the host for which the alarm is generated.

Step 2 Log in to the host where the alarm is generated as user **root**.

Step 3 Run the **vi ${BIGDATA_LOG_HOME}/controller/scriptlog/getLogs.log** command to check whether the keyword "LOG SIZE is more than 5000MB" can be searched.

- If it can, go to **Step 4**.
- If it cannot, go to **Step 5**.

Step 4 Check whether the oversized service audit logs are caused by exceptions.

The OMS backup storage space is insufficient.

Step 5 Run the **vi ${BIGDATA_LOG_HOME}/controller/scriptlog/getLogs.log** command to check whether the keyword "Collect log failed, too many logs on" can be searched.

- If it can, obtain the host IP address following the keyword "Collect log failed, too many logs on", and go to **Step 6**.
- If it cannot, go to **Step 10**.

Step 6 Log in to the host with the IP address obtained in as user **root**.

Step 7 Run the **vi {BIGDATA_LOG_HOME}/nodeagent/scriptlog/collectLog.log** command to check whether the keyword "log size exceeds" can be searched.

- If it can, go to **Step 8**.
- If it cannot, go to **Step 10**.

Step 8 Expand the capacity of the OMS node.

Step 9 In the next execution period, 03:00, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 10**.

Check whether the space of the host where the service is located is insufficient.

Step 10 Run the **vi ${BIGDATA_LOG_HOME}/controller/scriptlog/getLogs.log** command to check whether the keyword "Collect log failed, no enough space on *hostIp*" can be searched.

- If it can, obtain the IP address of the abnormal host and go to **Step 11**.
- If it cannot, go to **Step 14**.

Step 11 Log in to the host with the IP address obtained as user **root**, and run the **df "$BIGDATA_HOME/tmp" -lP | tail -1 | awk '{print ($4/1024)}'** command to obtain the remaining space of the host log directory. Check whether the value is less than 1000 MB.

- If it is, go to **Step 12**.
- If it is not, go to **Step 14**.

Step 12 Expand the capacity of the node

Step 13 In the next execution period, 03:00, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 14**.

Collect fault information.

Step 14 On FusionInsight Manager, choose **O&M**> **Log** > **Download**.

Step 15 Select **Controller** for **Service** and click **OK**.

Step 16 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 17 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.59 ALM-12087 System Is in the Upgrade Observation Period

Description

The system checks whether it is in the upgrade observation period at 00:00 every day and checks whether the duration that it has been in the upgrade observation state exceeds the preset upgrade observation period, 10 days by default. This alarm is generated when the system is in the upgrade observation period and the duration that the system has been in the upgrade observation state exceeds the preset period (10 days by default). This alarm is automatically cleared if the system exits the upgrade observation period after the user performs a rollback or submission.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12087	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Upgrade Observation Period (Days)	Specifies the days that the system is in the upgrade observation period.

Impact on the System

The next upgrade or patch installation will fail.

Possible Causes

The upgrade task is not submitted a specified period of time (10 days by default) after the system upgrade.

Procedure

Check whether the system is in the upgrade observation period.

Step 1 Log in to the active management node as user **root**.

Step 2 Run the following commands to switch to user **omm** and log in to the **omm** database:

su - omm

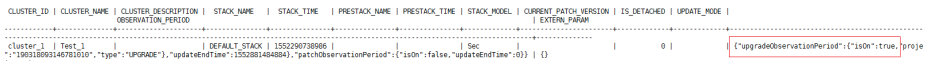
gsql -U omm -W *omm database password* **-p 20015**

Step 3 Run the **select * from OM_CLUSTERS** command to view cluster information.

Step 4 Check whether the value of **upgradObservationPeriod isON** is **true**, as shown in **Figure 10-33**.

- If it is, the system is in the upgrade observation period. Use the UpdateTool to submit the upgrade task. For details, see the upgrade guide of the corresponding version.
- If it is not, go to **Step 6**.

Figure 10-33 Cluster information



Step 5 In the early morning of the next day, check whether this alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **Controller** from the **Service** and click **OK**.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.60 ALM-12089 Inter-Node Network Is Abnormal

Description

The alarm module checks the network health status of nodes in the cluster every 10 seconds. This alarm is generated when the network between two nodes is unreachable or the network status is unstable.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12089	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Functions of some components, such as HDFS and ZooKeeper, are affected.

Possible Causes

- The node breaks down.
- The network is faulty.

Procedure

Check the network health status.

Step 1 In the alarm list on FusionInsight Manager, click the drop-down button of the alarm and view **Additional Information**. Record the source IP address and destination IP address of the node for which the alarm is reported.

Step 2 Log in to the node for which the alarm is reported. On the node, ping the target node to check whether the network between the two nodes is normal.

- If yes, go to **6**.

- If no, go to **3**.

Check the node status.

Step 3 On FusionInsight Manager, click **Host** and check whether the host list contains the faulty node to determine whether the faulty node has been removed from the cluster.

- If yes, go to **5**.
- If no, go to **4**.

Step 4 Check whether the faulty node is powered off.

- If yes, start the faulty node and go to **Step 2**.
- If no, contact related personnel to find root cause, if need to remove the faulty nodes from the cluster and go to **5**, otherwise go to **6**.

Step 5 Remove the file **$NODE_AGENT_HOME/etc/agent/hosts.ini** of all nodes in the cluster, and clean up the file **/var/log/Bigdata/unreachable/unreachable_ip_info.log**, and then manually clear the alarm.

Step 6 Wait for 30 seconds and checking if the alarm was been cleared.

- If yes, no further action is required.
- If no, go to **7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 8 Select **OmmAgent** from the **Service** and click **OK**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.61 ALM-12101 AZ Unhealthy

Description

After the AZ DR function is enabled, the system checks the AZ health status every 5 minutes. This alarm is generated when the system detects that the AZ is subhealthy or unhealthy. This alarm is cleared when the AZ becomes healthy.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12101	Major	Yes

Parameters

Parameter	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
AZName	Specifies the AZ for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The health status of an AZ is determined by whether the health status of storage resources (HDFS), computing resources (Yarn), and key roles in the AZ exceeds the configured threshold.

An AZ is subhealthy when:

- The computing resources (Yarn) are unhealthy, but the storage resources (HDFS) are healthy. Tasks cannot be submitted to the local AZ, but data can still be read and written in the local AZ.
- The computing resources (Yarn) are healthy, but some storage resources (HDFS) are unhealthy. Tasks can be submitted to the local AZ, and some data can be read and written in the local AZ. This depends on the locality of data detected by Spark/Hive scheduling.

An AZ is unhealthy when:

- The computing resources (Yarn) are healthy, but the storage resources (HDFS) are unhealthy. Although tasks can be submitted to the local AZ, data cannot be read or written in the local AZ. As a result, the tasks submitted to the local AZ are invalid.
- The computing resources (Yarn) and storage resources (HDFS) are unhealthy. Tasks cannot be submitted to the local AZ, and data cannot be read or written in the local AZ.
- The health status of key roles except Yarn and HDFS is lower than the configured threshold.

Possible Causes

- The computing resources (Yarn) are unhealthy.
- The storage resources (HDFS) are unhealthy.
- Some storage resources (HDFS) are unhealthy.
- Key roles except Yarn and HDFS are unhealthy.

Procedure

Disable the DR drill.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Cross-AZ HA**. The Cross-AZ HA page is displayed.

Step 2 In the AZ DR list, check whether **Perform DR Drill** in the **Operation** column of the AZ whose health status is **Unhealthy** is gray.

- If yes, go to **Step 4**.
- If no, go to **Step 3**.

Step 3 Click **Restore** in the **Operation** column of the target AZ. Wait 2 minutes and refresh the page to view the health status of the AZ. Check whether the health status is normal.

- If yes, no further action is required.
- If no, go to **Step 4**.

Collect the fault information.

Step 4 Log in to the active management node as user **root**.

Step 5 View logs of unhealthy services.

- HDFS log files are stored in **/var/log/Bigdata/hdfs/nn/hdfs-az-state.log**.
- Yarn log files are stored in **/var/log/Bigdata/yarn/rm/yarn-az-state.log**.
- For other services, view the service health check logs in the corresponding service log directory.

Step 6 Contact O&M personnel and provide detailed log file information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.62 ALM-12102 AZ HA Component Is Not Deployed Based on DR Requirements

Description

The alarm module checks the deployment status of AZ HA components every 5 minutes. This alarm is generated when the components that support DR are not

deployed based on DR requirements after AZ is enabled. This alarm is cleared when the components are deployed based on DR requirements.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12102	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Impact on the System

The cross-AZ HA capability of a single cluster is affected.

Possible Causes

The roles of the components that support DR are not deployed based on DR requirements.

Procedure

Obtain alarm information.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**.

Step 2 In the alarm list, click ⌄ in the row that contains the alarm and view the roles that are not deployed based on DR requirements in **Additional Information**.

Redeploy the role instance.

Step 3 Choose **Cluster** > **Services** > *Name of the desired service* > **Instance**. On the instance page, redeploy or adjust the role instance.

Step 4 Check whether the alarm is cleared 10 minutes later.

- If yes, no further action is required.
- If no, contact O&M personnel.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.63 ALM-12110 Failed to get ECS temporary ak/sk

Description

The meta service periodically obtains the temporary AK/SK of the ECS. This alarm is generated when the meta service fails to obtain the temporary AK/SK.

Attribute

Alarm ID	Alarm Severity	Auto Clear
12110	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

In storage-compute decoupling scenarios, the cluster cannot obtain the latest temporary AK/SK, which may lead to failure access to OBS.

Possible Causes

- The meta role of the MRS cluster is abnormal.
- The cluster has been bound to an agency and accessed OBS but has been unbound from the agency. As a result, the cluster has not been bound to any agency.

Procedure

Check the status of the meta role.

Step 1 On FusionInsight Manager of the cluster, choose **O&M** > **Alarm** > **Alarms**. On the page that is displayed, click ⌄ in the row containing the alarm, and determine the IP address of the host for which the alarm is generated.

Step 2 On FusionInsight Manager of the cluster, choose **Cluster** > **Services** > **meta**. On the page that is displayed, click the **Instance** tab, and check whether the meta role corresponding to the host for which the alarm is generated is normal.

- If yes, go to **Step 4**.
- If no, go to **Step 3**.

Step 3 Select the abnormal role and choose **More** > **Restart Instance** to restart the abnormal meta role. After the restart is complete, check whether the alarm is cleared several minutes later.

- If yes, no further action is required.
- If no, go to **Step 4**.

Rebind the cluster to an agency.

Step 4 Log in to the MRS management console.

Step 5 In the navigation pane on the left, choose **Clusters** > **Active Clusters**. On the page that is displayed, click the cluster name to go to its overview page. Then, check whether the cluster is bound to an agency.

- If yes, go to **Step 7**.
- If no, go to **Step 6**.

Step 6 Click **Manage Agency**. On the page that is displayed, rebind the cluster to an agency. Then check whether the alarm is cleared a few minutes later.

- If yes, no further action is required.
- If no, go to **Step 7**.

Step 7 Contact O&M personnel.

----End

10.13.64 ALM-13000 ZooKeeper Service Unavailable

Description

The system checks the ZooKeeper service status every 60 seconds. This alarm is generated when the ZooKeeper service is unavailable.

This alarm is cleared when the ZooKeeper service recovers.

Attribute

Alarm ID	Alarm Severity	Auto Clear
13000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

ZooKeeper cannot provide coordination services for upper layer components and the components that depend on ZooKeeper may not run properly.

Possible Causes

- The DNS is installed on the ZooKeeper node.
- The network is faulty.
- The KrbServer service is abnormal.
- The ZooKeeper instance is abnormal.
- The disk capacity is insufficient.

Procedure

Check the DNS.

Step 1 Check whether the DNS is installed on the node where the ZooKeeper instance is located. On the Linux node where the ZooKeeper instance is located, run the **cat /etc/resolv.conf** command to check whether the file is empty.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Run the **service named status** command to check whether the DNS is started.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Run the **service named stop** command to stop the DNS service. If "Shutting down name server BIND waiting for named to shut down (28s)" is displayed, the DNS service is stopped successfully. Comment out the content (if any) in **/etc/resolv.conf**.

Step 4 On the **O&M > Alarm > Alarms** tab, check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 5**.

Check the network status.

Step 5 On the Linux node where the ZooKeeper instance is located, run the **ping** command to check whether the host names of other nodes where the ZooKeeper instance is located can be pinged successfully.

- If yes, go to **Step 9**.
- If no, go to **Step 6**.

Step 6 Modify the IP addresses in **/etc/hosts** and add the host name and IP address mapping.

Step 7 Run the **ping** command again to check whether the host names of other nodes where the ZooKeeper instance is located can be pinged successfully.

- If yes, go to **Step 8**.
- If no, go to **Step 23**.

Step 8 On the **O&M > Alarm > Alarms** tab, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Check the KrbServer service status (Skip this step if the normal mode is used).

Step 9 On FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Services**.

Step 10 Check whether the KrbServer service is normal.

- If yes, go to **Step 13**.
- If no, go to **Step 11**.

Step 11 Perform operations based on "ALM-25500 KrbServer Service Unavailable" and check whether the KrbServer service is recovered.

- If yes, go to **Step 12**.
- If no, go to **Step 23**.

Step 12 On the **O&M > Alarm > Alarms** tab, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Check the ZooKeeper service instance status.

Step 13 On FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **quorumpeer**.

Step 14 Check whether the ZooKeeper instances are normal.

- If yes, go to **Step 18**.
- If no, go to **Step 15**.

Step 15 Select instances whose status is not good, and choose **More** > **Restart Instance**.

Step 16 Check whether the instance status is good after restart.

- If yes, go to **Step 17**.
- If no, go to **Step 18**.

Step 17 On the **O&M > Alarm > Alarms** tab, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Check disk status.

Step 18 On FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Service** > **ZooKeeper** > **quorumpeer**, and check the node host information of the ZooKeeper instance.

Step 19 On FusionInsight Manager, click **Host**.

Step 20 In the **Disk** column, check whether the disk space of each node where ZooKeeper instances are located is insufficient (disk usage exceeds 80%).

- If yes, go to **Step 21**.
- If no, go to **Step 23**.

Step 21 Expand disk capacity. For details, see "ALM-12017 Insufficient Disk Capacity".

Step 22 On the **O&M > Alarm > Alarms** tab, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 23**.

Collect fault information.

Step 23 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 24 Select the following nodes in the required cluster from the **Service**: (KrbServer logs do not need to be downloaded in normal mode.)

- ZooKeeper
- KrbServer

Step 25 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 26 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.65 ALM-13001 Available ZooKeeper Connections Are Insufficient

Description

The system checks ZooKeeper connections every 60 seconds. This alarm is generated when the system detects that the number of used ZooKeeper instance connections exceeds the threshold (80% of the maximum connections).

When the **Trigger Count** is 1, this alarm is cleared when the number of used ZooKeeper instance connections is smaller than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the number of used ZooKeeper instance connections is smaller than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
13001	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host name for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Available ZooKeeper connections are insufficient. When the connection usage reaches 100%, external connections cannot be handled.

Possible Causes

The number of connections to the ZooKeeper node exceeds the threshold. Connection leakage occurs on some connection processes, or the maximum number of connections does not comply with the actual scenario.

Procedure

Check connection status.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **Available ZooKeeper Connections Are Insufficient** and confirm the node IP address of the host for which the alarm is generated in the Location Information.

Step 2 Obtain the PID of the ZooKeeper process. Log in to the node involved in this alarm as user **root** and run the **pgrep -f proc_zookeeper** command.

Step 3 Check whether the PID can be correctly obtained.

- If yes, go to **Step 4**.
- If no, go to **Step 15**.

Step 4 Obtain all the IP addresses connected to the ZooKeeper instance and the number of connections and check 10 IP addresses with top connections. Run the following command based on the obtained PID: **lsof -i|grep** *$pid* **| awk '{print $9}' | cut -d : -f 2 | cut -d \>-f 2 | awk '{a[$1]++} END {for(i in a){print i,a[i] | "sort -r -g -k 2"}}' | head -10**. (The PID obtained in the preceding step is used.)

Step 5 Check whether node IP addresses and number of connections are successfully obtained.

- If yes, go to **Step 6**.
- If no, go to **Step 15**.

Step 6 Obtain the ID of the port connected to the process. Run the following command based on the obtained PID and IP address: **lsof -i|grep** *$pid* **| awk '{print $9}'|cut -d \> -f 2 |grep** *$IP*| **cut -d : -f 2**. (The PID and IP address obtained in the preceding step are used.)

Step 7 Check whether the port ID is successfully obtained.

- If yes, go to **Step 8**.
- If no, go to **Step 15**.

Step 8 Obtain the ID of the connected process. Log in to each IP address and run the following command based on the obtained port ID: **lsof -i|grep** *$port*. (The port ID obtained in the preceding step is used.)

Step 9 Check whether the process ID is successfully obtained.

- If yes, go to **Step 10**.
- If no, go to **Step 15**.

Step 10 Check whether connection leakage occurs on the process based on the obtained process ID.

- If yes, go to **Step 11**.

- If no, go to **Step 12**.

Step 11 Close the process where connection leakage occurs and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Step 12 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Configurations** > **All Configurations** > **quorumpeer** > **Performance** and increase the value of **maxCnxns** as required.

Step 13 Save the configuration and restart the ZooKeeper service.

Step 14 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Collect fault information.

Step 15 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 16 Select **ZooKeeper** in the required cluster from the **Service**:

Step 17 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact the O&M personnel and send the collected log information.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.66 ALM-13002 ZooKeeper Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the ZooKeeper service every 30 seconds. The alarm is generated when the direct memory usage of a ZooKeeper instance exceeds the threshold (80% of the maximum memory).

When the **Trigger Count** is 1, this alarm is cleared when the ZooKeeper Direct memory usage is less than the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the ZooKeeper Direct memory usage is less than 80% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
13002	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the ZooKeeper service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the ZooKeeper instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **ZooKeeper Direct Memory Usage Exceeds the Threshold**. Check the IP address of the instance that reports the alarm.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Instance** > **quorumpeer(the IP address checked)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **CPU and Memory**, and select **ZooKeeper Heap And Direct Buffer Resource Percentage**, click **OK**.

Step 3 Check whether the used direct buffer memory of ZooKeeper reaches 80% of the maximum direct buffer memory specified for ZooKeeper.

- If yes, go to **Step 4**.
- If no, go to **Step 8**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Configurations** > **All Configurations** > **quorumpeer** > **System** to check whether "-XX:MaxDirectMemorySize" exists in the **GC_OPTS** parameter.

- If yes, in the **GC_OPTS** parameter, delete "-XX:MaxDirectMemorySize" and go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Save the configuration and restart the ZooKeeper service.

Step 6 Check whether the **ALM-13004 ZooKeeper Heap Memory Usage Exceeds the Threshold** exists.

- If yes, handle the alarm by referring to **ALM-13004 ZooKeeper Heap Memory Usage Exceeds the Threshold**.
- If no, go to **Step 7**.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 9 Select **ZooKeeper** in the required cluster from the **Service**.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.67 ALM-13003 GC Duration of the ZooKeeper Process Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the ZooKeeper process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default).

This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
13003	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the ZooKeeper process may interrupt the services.

Possible Causes

The heap memory of the ZooKeeper instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **GC Duration of the ZooKeeper Process Exceeds the Threshold**. Check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Instance** > **quorumpeer(IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **GC**, and select **ZooKeeper GC Duration per Minute**, click **OK** to check the GC duration statistics of the ZooKeeper process collected every minute.

Step 3 Check whether the GC duration of the ZooKeeper process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 8**.

Step 4 Check whether memory leakage occurs in the application program.

Step 5 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Configurations** > **All Configurations** > **quorumpeer** > **System** to increase the value of **GC_OPTS** parameter as required.

NOTE

Generally, the value of **-Xmx** is twice of the ZooKeeper data capacity. You are advised to set **MaxDirectMemorySize** to half of the data capacity. If the ZooKeeper capacity reaches 2G, you are advised to set GC_OPTS to:

-Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=512M -XX:MetaspaceSize=64M -XX:MaxMetaspaceSize=64M -XX:CMSFullGCsBeforeCompaction=1

Step 6 Save the configuration and restart the ZooKeeper service.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 9 Select **ZooKeeper** in the required cluster from the **Service**.

Step 10 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.68 ALM-13004 ZooKeeper Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of the ZooKeeper service every 60 seconds. The alarm is generated when the heap memory usage of a ZooKeeper instance exceeds the threshold (95% of the maximum memory).

The alarm is cleared when the memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
13004	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available ZooKeeper heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The heap memory of the ZooKeeper instance is overused or the heap memory is inappropriately allocated.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, On the displayed interface, click the drop-down button of **ZooKeeper Heap Memory Usage Exceeds the Threshold** and confirm the node IP address of the host for which the alarm is generated in the Location Information.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Instance**, click **quorumpeer** in the **Role** column of the corresponding IP address. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **CPU and Memory**, and select **ZooKeeper Heap And Direct Buffer Resource Percentage**, click **OK**. Check the heap memory usage.

Step 3 Check whether the used heap memory of ZooKeeper reaches 95% of the maximum heap memory specified for ZooKeeper.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Configurations** > **All Configurations** > **quorumpeer** > **System**. Increase the value of **-Xmx** in **GC_OPTS** as required. The details are as follows:

1. On the **Instance** tab, click **quorumpeer** in the **Role** column of the corresponding IP address. Choose **Customize** > **CPU and Memory** in the upper right corner, and select **ZooKeeper Heap And Direct Buffer Resource**, click **OK** to check the heap memory used by ZooKeeper.
2. Change the value of **-Xmx** in the **GC_OPTS** parameter based on the actual heap memory usage. Generally, the value is twice the size of the ZooKeeper data volume. For example, if 2 GB ZooKeeper heap memory is used, the following configurations are recommended: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=512M -XX:MetaspaceSize=64M -XX:MaxMetaspaceSize=64M -XX:CMSFullGCsBeforeCompaction=1

Step 5 Save the configuration and restart the ZooKeeper service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **ZooKeeper** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.69 ALM-13005 Failed to Set the Quota of Top Directories of ZooKeeper Components

Description

The system sets quotas for each ZooKeeper top-level directory in the **customized.quota** configuration item and components every 5 hours. This alarm is generated when the system fails to set the quota for a directory.

This alarm is cleared when the setting succeeds after a failure.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
13005	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
ServiceDirectory	Specifies the directory for which the alarm is generated.
Trigger Condition	Specifies the cause of the alarm.

Impact on the System

Components can write a large amount of data to the top-level directory of ZooKeeper. As a result, the ZooKeeper service is unavailable.

Possible Causes

The quota for the alarm directory is inappropriate.

Procedure

Check whether the quota for the alarm directory is appropriate.

Step 1 Log in to FusionInsight Manager, and choose **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper**. On the displayed page, choose **Configurations** > **All Configurations** > **Quota**. Check whether the directory for which the alarm is reported and its quota exist in the **customized.quota** configuration item.

- If yes, go to **Step 5**.
- If no, go to **Step 2**.

Step 2 Check whether the alarm directory for which the alarm is reported is in the following alarm list.

Table 10-87 Component alarm directory

Component	Alarm Directory
Hbase	/hbase
Hive	/beelinesql
Yarn	/rmstore
Storm	/stormroot
Streaming	/storm
Kafka	/kafka

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 View the component of the alarm directory in the table, open the corresponding service page, and choose **Configurations** > **All Configurations**. On the displayed page, search for **zk.quota** in the upper right corner. The search result is the quota of the alarm directory.

Step 4 Check whether the quota of the alarm directory for which the alarm is reported is appropriate. The quota must be greater than or equal to the actual value, which can be obtained in **Trigger Condition**.

Step 5 Modify the **services.quota** value as prompted and save the configuration.

Step 6 After the time specified by **service.quotas.auto.check.cron.expression**, check whether the alarm is cleared.

- If it is, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **ZooKeeper** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.70 ALM-13006 Znode Number or Capacity Exceeds the Threshold

Description

The system periodically detects the status of secondary Znode in the ZooKeeper service data directory every four hours. This alarm is generated when the number or capacity of secondary Znodes exceeds the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
13006	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
ServiceDirectory	Specifies the directory for which the alarm is generated.
Trigger Condition	Specifies the cause of the alarm.

Impact on the System

A large amount of data is written to the ZooKeeper data directory. As a result, ZooKeeper cannot provide normal services.

Possible Causes

A large amount of data is written to the ZooKeeper data directory. The threshold is not appropriate.

Procedure

Check whether a large amount of data is written to the directory for which the alarm is generated.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **Znode Number or Capacity Exceeds the Threshold**. Confirm the Znode for which the alarm is generated in Location Information.

Step 2 Log in to FusionInsight Manager, open the ZooKeeper service interface, and select **Resource**. In the table **Used Resources (By Second-Level Znode)**, check whether a large amount of data is written to the top-level Znode for which the alarm is reported.

- If it is, go to **Step 3**.
- If it is not, go to **Step 4**.

Step 3 Log in to the ZooKeeper client and delete the data in the top-level Znode.

Step 4 Log in to FusionInsight Manager and open the ZooKeeper service interface. On the **Resource** page, choose > **By Znode quantity** in **Used Resources (By Second-Level Znode)**. **Threshold Configuration of By Znode quantity** is displayed. Click **Modify** under **Operation**. Increase the threshold by referringto the value of **max.Znode.count** by choosing **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Configurations > All Configurations > Quota**.

Step 5 In the **Used Resources (By Second-Level Znode)**, choose > **By capacity**. The **Threshold Settings** page of **By Capacity** is displayed. Click **Modify** under **Operation**. Increase the threshold by referring to the value of **max.data.size** by choosing **Cluster >** *Name of the desired cluster* **> Services** > **ZooKeeper** > **Configurations > All Configurations** > **Quota**.

Step 6 Check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **ZooKeeper** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.71 ALM-13007 Available ZooKeeper Client Connections Are Insufficient

Description

The system periodically detects the number of active processes between the ZooKeeper client and the ZooKeeper server every 60 seconds. This alarm is generated when the number of connections exceeds the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
13007	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the host name for which the alarm is generated.
ClientIP	Specifies the client IP address.
ServerIP	Specifies the server IP address.
Trigger Condition	Specifies the cause of the alarm.

Impact on the System

A large number of connections to ZooKeeper caused the ZooKeeper to be fully connected and unable to provide normal services.

Possible Causes

A large number of client processes are connected to ZooKeeper. The thresholds are not appropriate.

Procedure

Check whether there are a large number of client processes connected to ZooKeeper.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **Available ZooKeeper Client Connections Are Insufficient**. Confirm the node IP address of the host for which the alarm is generated in the Location Information.

Step 2 Open the ZooKeeper service interface, click **Resource** to enter the **Resource** page, and check whether the number of connections of the client with the IP address specified by **Number of Connections (By Client IP Address)** is large.

- If it is, go to **Step 3**.
- If it is not, go to **Step 4**.

Step 3 Check whether connection leakage occurs on the client process.

Step 4 Click in the **Number of Connections (by Client IP Address)** to enter the **Thresholds** page, and click **Modify** under **Operation**. Increase the threshold by referring to the value of **maxClientCnxns** by choosing **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Configurations** > **All Configurations** > **quorumpeer**.

Step 5 Check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **ZooKeeper** in the required cluster from the **Service**.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.72 ALM-13008 ZooKeeper Znode Usage Exceeds the Threshold

Description

The system checks the level-2 Znode status in the ZooKeeper data directory every hour. This alarm is generated when the system detects that the level-2 Znode usage exceeds the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
13008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
ServiceDirectory	Specifies the directory for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the cause of the alarm.

Impact on the System

A large amount of data is written to the ZooKeeper data directory. As a result, ZooKeeper cannot provide services properly.

Possible Causes

- A large amount of data is written to the ZooKeeper data directory.
- The user-defined threshold is inappropriate.

Procedure

Check whether a large amount of data is written into the directory for which the alarm is generated.

Step 1 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper**, and click **Resource**. Click **By Znode quantity** in **Used**

Resources (By Second-Level Znode), and check whether a large amount of data is written to the top Znode.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Log in to FusionInsight Manager, choose **O&M > Alarm > Alarms**, select **Location** from the drop-down list box next to **ALM-13008 ZooKeeper Znode Quantity Usage Exceeds Threshold**, and obtain the Znode path in **ServiceDirectory**.

Step 3 Log in to the ZooKeeper client as a cluster user and delete unnecessary data from the Znode corresponding to the alarm.

Step 4 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Configurations** > **All Configurations**, and search for **max.znode.count**, which is the maximum number of ZooKeeper directories. The alarm threshold is 80% of this parameter. Increase the value of this parameter, click **Save**, and restart the service for the configuration to take effect.

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 7 Select **ZooKeeper** in the required cluster from the **Service**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.73 ALM-13009 ZooKeeper Znode Capacity Usage Exceeds the Threshold

Description

The system checks the level-2 Znode status in the ZooKeeper data directory every hour. This alarm is generated when the system detects that the capacity usage exceeds the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
13009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
ServiceDirectory	Specifies the directory for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the cause of the alarm.

Impact on the System

A large amount of data is written to the ZooKeeper data directory. As a result, ZooKeeper cannot provide services properly.

Possible Causes

- A large amount of data is written to the ZooKeeper data directory.
- The user-defined threshold is inappropriate.

Procedure

Check whether a large amount of data is written into the directory for which the alarm is generated.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. Confirm the Znode for which the alarm is generated in **Location** of this alarm.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** and click **Resource**. Click **By capacity** in **Used Resources (By Second-Level Znode)** and check whether a large amount of data is written into the top Znode directory.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Log in to FusionInsight Manager, choose **O&M > Alarm > Alarms**, select **Location** from the drop-down list box next to **ALM-13009 ZooKeeper Znode Capacity Usage Exceeds the Threshold**, and obtain the Znode path in **ServiceDirectory**.

Step 4 Log in to the ZooKeeper client as a cluster user and delete unwanted data in the Znode for which the alarm is generated.

Step 5 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Configurations** > **All Configurations**, and search for **max.data.size**, which is the maximum capacity quota of the ZooKeeper directory in bytes. Search for the **GC_OPTS** configuration item and check the value of **Xmx**.

Step 6 Compare the values of **max.data.size** and **Xmx** multiplied by 0.65. The threshold is the smaller value multiplied by 80%. You can change the values of **max.data.size** and **Xmx** to increase the threshold.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 9 Select **ZooKeeper** in the required cluster from the **Service**.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.74 ALM-13010 Znode Usage of a Directory with Quota Configured Exceeds the Threshold

Description

The system checks the Znode usage of all service directories with quota configured every hour. This alarm is generated when the system detects that the level-2 Znode usage exceeds the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
13010	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
ServiceDirectory	Specifies the directory for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the cause of the alarm.

Impact on the System

A large amount of data is written to the ZooKeeper data directory. As a result, ZooKeeper cannot provide services properly.

Possible Causes

- A large amount of data is written to the ZooKeeper data directory.
- The user-defined threshold is inappropriate.

Procedure

Check whether a large amount of data is written into the directory for which the alarm is generated.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. Confirm the Znode for which the alarm is generated in **Location** of this alarm.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** and click **Resource**. In **Used Resources (By Second-Level Znode)**, check whether a large amount of data is written into the top Znode.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 3 Log in to FusionInsight Manager, choose **O&M > Alarm > Alarms**, select Location from the drop-down list box next to **ALM-13010 Znode Usage of a Directory with Quota Configured Exceeds the Threshold**, and obtain the Znode path in ServiceDirectory.

Step 4 Log in to the ZooKeeper client as a cluster user and delete unwanted data in the Znode for which the alarm is generated.

Step 5 Log in to FusionInsight Manager, and choose **Cluster** > *Name of the desired cluster* > **Services** > *Component of the top Znode for which the alarm isgenerated*. Choose **Configurations** > **All Configurations**, search for **zk.quota.number**, increase its value, click **Save**.

NOTICE

If the Component of the top Znode for which the alarm isgenerated is ClickHouse, change the value of **clickhouse.zookeeper.quota.node.count**.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 8 Select **ZooKeeper** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.75 ALM-14000 HDFS Service Unavailable

Description

The system checks the NameService service status every 60 seconds. This alarm is generated when all the NameService services are abnormal and the system considers that the HDFS service is unavailable.

This alarm is cleared when at least one NameService service is normal and the system considers that the HDFS service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

HDFS fails to provide services for HDFS service-based upper-layer components, such as HBase and MapReduce. As a result, users cannot read or write files.

Possible Causes

- The ZooKeeper service is abnormal.
- All NameService services are abnormal.

Procedure

Check the ZooKeeper service status.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. On the Alarm page, check whether **ALM-13000 ZooKeeper Service Unavailable** is reported.

- If yes, go to **Step 2**.
- If no, go to **Step 4**.

Step 2 See **ALM-13000 ZooKeeper Service Unavailable** to rectify the health status of ZooKeeper fault and check whether the **Running Status** of the ZooKeeper service restores to **Normal**.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the **O&M > Alarm > Alarms** page, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Handle the NameService service exception alarm.

Step 4 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. On the Alarms page, check whether **ALM-14010 NameService Service Unavailable** is reported.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 See **ALM-14010 NameService Service Unavailable** to handle the abnormal NameService services and check whether each NameService service exception alarm is cleared.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 6 On the **O&M > Alarm > Alarms** page, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select the following nodes in the required cluster from the **Service**:

- ZooKeeper
- HDFS

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.76 ALM-14001 HDFS Disk Usage Exceeds the Threshold

Description

The system checks the HDFS disk usage every 30 seconds and compares the actual HDFS disk usage with the threshold. The HDFS disk usage indicator has a default threshold, this alarm is generated when the value of the disk usage of a Hadoop distributed file system (HDFS) indicator exceeds the threshold.

To change the threshold, choose **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* **> HDFS**.

When the **Trigger Count** is 1, this alarm is cleared when the value of the disk usage of HDFS cluster indicator is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the value of the disk usage of HDFS cluster indicator is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14001	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Writing Hadoop distributed file system (HDFS) data is affected.

Possible Causes

The disk space configured for the HDFS cluster is insufficient.

Procedure

Check the disk capacity and delete unnecessary files.

Step 1 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS**.

Step 2 Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Disk**, and select **Percentage of HDFS Capacity** to check whether the HDFS disk usage exceeds the threshold (80% by default).

- If yes, go to **Step 3**.
- If no, go to **Step 11**.

Step 3 In the **Basic Information** area, click the **NameNode(Active)** of the failure NameService and the HDFS WebUI page is displayed.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 4 On the HDFS web user interface (WebUI), click **Datanodes** tab. In the **Block pool used** column, view the disk usage of all DataNodes to check whether the disk usage of any DataNode exceeds the threshold.

- If yes, go to **Step 6**.
- If no, go to **Step 11**.

Step 5 Log in to the MRS client node as user **root**.

Step 6 Run **cd /opt/Bigdata/client** to switch to the client installation directory, and run **source bigdata_env**. If the cluster uses the security mode, perform security authentication. Run **kinit hdfs** and enter the password as prompted. Please obtain the password from the administrator.

Step 7 Run the **hdfs dfs -rm -r** *file or directory* command to delete unnecessary files.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Expand the system.

Step 9 Expand the disk capacity.

Step 10 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 12 Select the following nodes in the required cluster from the **Service**:

- ZooKeeper
- HDFS

Step 13 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.77 ALM-14002 DataNode Disk Usage Exceeds the Threshold

Description

The system checks the disk usage of the DataNode every 30 seconds and compares the actual disk usage with the threshold. The DataNode Disk Usage indicator has a default threshold. This alarm is generated when the value of the DataNode Disk Usage indicator exceeds the threshold.

To change the threshold, choose **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* **> HDFS**.

When the **Trigger Count** is 1, this alarm is cleared when the value of the DataNode Disk Usage indicator is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the value of the DataNode Disk Usage indicator is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14002	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Writing Hadoop distributed file system (HDFS) data is affected.

Possible Causes

- The cluster disk space is full.
- Data among DataNode nodes is skew.

Procedure

Check whether the cluster disk space is full.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms**, check whether the **ALM-14001 HDFS Disk Usage Exceeds the Threshold** alarm exists.

- If yes, run **Step 2**.
- If no, run **Step 4**.

Step 2 Handle the alarm as instructed in **ALM-14001 HDFS Disk Usage Exceeds the Threshold**. Check whether the alarm is cleared.

- If yes, run **Step 3**.
- If no, run **Step 11**.

Step 3 On the **O&M > Alarm > Alarms** pages, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, run **Step 4**.

Check the balancing status of DataNode nodes.

Step 4 On the FusionInsight Manager portal, click **Host**. Check the number of DataNodes on each rack. If the number differs greatly, adjust the racks to ensure that the number of DataNodes on each rack is almost the same. Restart the HDFS service for the changes to take effect.

Step 5 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**.

Step 6 In the **Basic Information** area, click **NameNode(Active)** and the HDFS WebUI page is displayed.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 7 In the **Summary** area of HDFS WebUI, check whether the value of **Max** is 10% greater than that of **Median** in **DataNodes usages**.

- If yes, go to **Step 8**.
- If no, go to **Step 11**.

Step 8 Data in the cluster is skew and must be balanced. Log in to the MRS client as user **root**. If the cluster uses the Normal Mode, run **su - omm** to switch to user **omm**. Run **cd** to switch to the client installation directory, and run **source bigdata_env**. If the cluster uses the security mode, perform security authentication. Run **kinit**

hdfs and enter the password as prompted. Please obtain the password from the administrator.

Step 9 Run the following command to balance the data distribution:

hdfs balancer -threshold 10

Step 10 Wait several minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 12 Select **HDFS** in the required cluster from the **Service**.

Step 13 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.78 ALM-14003 Number of Lost HDFS Blocks Exceeds the Threshold

Description

The system checks the lost blocks every 30 seconds and compares the actual lost blocks with the threshold. The lost blocks indicator has a default threshold. This alarm is generated when the number of lost HDFS blocks exceeds the threshold.

To change the threshold, choose **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* **> HDFS**.

When the **Trigger Count** is 1, this alarm is cleared when the value of lost HDFS blocks is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the value of lost HDFS blocks is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
14003	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Data stored in HDFS is lost. HDFS may enter the safe mode and cannot provide write services. Lost block data cannot be restored.

Possible Causes

- The DataNode instance is abnormal.
- Data is deleted.

Procedure

Check the DataNode instance.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Instance**.

Step 2 Check whether the **Running Status** of all DataNode instance is **Normal**.

- If yes, go to **Step 11**.
- If no, go to **Step 3**.

Step 3 Restart the DataNode instance and check whether the DataNode instance restarts successfully.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Delete the damaged file.

Step 5 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **NameNode(Active)**. On the WebUI page of the HDFS, view the information about lost blocks.

NOTE

- If a block is lost, a line in red is displayed on the WebUI.
- By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 6 The user checks whether the file containing the lost data block is useful.

NOTE

Files generated in directories **/mr-history**, **/tmp/hadoop-yarn**, and **/tmp/logs** during MapReduce task execution are unnecessary.

- If yes, go to **Step 7**.
- If no, go to **Step 8**.

Step 7 The user checks whether the file containing the lost data block is backed up.

- If yes, go to **Step 8**.
- If no, go to **Step 11**.

Step 8 Log in to the HDFS client as user **root**. The user password is defined by the user before the installation. Contact the system administrator to obtain the password. Run the following commands:

- Security mode:

  cd *Client installation directory*

  source bigdata_env

  kinit hdfs

- Normal mode:

  su - omm

  cd *Client installation directory*

  source bigdata_env

Step 9 On the node client, run **hdfs fsck / -delete** to delete the lost file. If the file where the lost block is located is a useful file, you need to write the file again to restore the data.

NOTE

Deleting a file or folder is a high-risk operation. Ensure that the file or folder is no longer required before performing this operation.

Step 10 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect the fault information.

Step 11 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 12 Select **HDFS** in the required cluster from the **Service**.

Step 13 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.79 ALM-14006 Number of HDFS Files Exceeds the Threshold

Description

The system periodically checks the number of HDFS files every 30 seconds and compares the number of HDFS files with the threshold. This alarm is generated when the system detects that the number of HDFS files exceeds the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the number of HDFS files is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the number of HDFS files is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
14006	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Disk storage space is insufficient, which may result in data import failure. The performance of the HDFS system is affected.

Possible Causes

The number of HDFS files exceeds the threshold.

Procedure

Check the number of files in the system.

Step 1 On FusionInsight Manager, check the number of HDFS files. Specifically, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **File and Block**, and select **HDFS File** and **Total Blocks**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations** > **All Configurations**, and search for the **GC_OPTS** parameter under **NameNode**.

Step 3 Configure the threshold of the number of configuration file objects. Specifically, change the value of **Xmx** (GB) in the **GC_OPTS** parameter. The threshold (specified by y) is calculated as follows: y = 0.2007 x Xmx - 0.6312, where x indicates the memory capacity Xmx (GB) and y indicates the number of files (unit: kW). Adjust the memory size as required.

Step 4 Confirm that the value of **GC_PROFILE** is **custom** so that the **GC_OPTS** configuration takes effect. Click **Save** and choose **More** > **Restart Instance** to restart the service.

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check whether needless files exist in the system.

Step 6 Log in to the HDFS client as user **root**. Run **cd** to switch to the client installation directory, and run **source bigdata_env** to configure the environment variables.

If the cluster uses the security mode, perform security authentication.

Run the **kinit hdfs** command and enter the password as prompted. Obtain the password from the administrator.

Step 7 Run **hdfs dfs -ls** *file or directory* to check whether the files in the directory can be deleted.

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Run the **hdfs dfs -rm -r** *file or directory path* command. After deleting unnecessary files, wait until the files are retained in the recycle bin for a period longer than the value of **fs.trash.interval** on the NameNode. Then check whether the alarm is cleared.

NOTE

Deleting a file or folder is a high-risk operation. Ensure that the file or folder is no longer required before performing this operation.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 Select **HDFS** in the required cluster from the **Service**.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

Configuration rules of the NameNode JVM parameter

Default value of the NameNode JVM parameter **GC_OPTS**:

-Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=128M -

XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -Djdk.tls.ephemeralDHKeySize=3072 -Djdk.tls.rejectClientInitiatedRenegotiation=true -Djava.io.tmpdir=${Bigdata_tmp_dir}

The number of NameNode files is proportional to the used memory size of the NameNode. When file objects change, you need to change **-Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M** in the default value. The following table lists the reference values.

Table 10-88 NameNode JVM configuration

Number of File Objects	Reference Value
10,000,000	-Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
20,000,000	-Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G
50,000,000	-Xms32G -Xmx32G -XX:NewSize=3G -XX:MaxNewSize=3G
100,000,000	-Xms64G -Xmx64G -XX:NewSize=6G -XX:MaxNewSize=6G
200,000,000	-Xms96G -Xmx96G -XX:NewSize=9G -XX:MaxNewSize=9G
300,000,000	-Xms164G -Xmx164G -XX:NewSize=12G -XX:MaxNewSize=12G

10.13.80 ALM-14007 NameNode Heap Memory Usage Exceeds the Threshold

Description

The system checks the HDFS NameNode Heap Memory usage every 30 seconds and compares the actual Heap memory usage with the threshold. The HDFS NameNode Heap Memory usage has a default threshold. This alarm is generated when the HDFS NameNode Heap Memory usage exceeds the threshold.

You can change the threshold in **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* **> HDFS**.

When the **Trigger Count** is 1, this alarm is cleared when the HDFS NameNode Heap memory usage is less than or equal to the threshold. When the **Trigger**

Count is greater than 1, this alarm is cleared when the HDFS NameNode Heap memory usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The HDFS NameNode Heap Memory usage is too high, which affects the data read/write performance of the HDFS.

Possible Causes

The HDFS NameNode Heap Memory is insufficient.

Procedure

Delete unnecessary files.

Step 1 Log in to the HDFS client as user **root**. Run **cd** to switch to the client installation directory, and run **source bigdata_env**.

If the cluster uses the security mode, perform security authentication.

Run the **kinit hdfs** command and enter the password as prompted. Obtain the password from the administrator.

Step 2 Run the **hdfs dfs -rm -r** *file or directory* command to delete unnecessary files.

Step 3 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check the NameNode JVM memory usage and configuration.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**.

Step 5 In the **Basic Information** area, click **NameNode(Active)** to go to the HDFS WebUI.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 6 On the HDFS WebUI, click the **Overview** tab. In **Summary**, check the numbers of files, directories, and blocks in the HDFS.

Step 7 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations** > **All Configurations**. In **Search**, enter **GC_OPTS** to check the **GC_OPTS** memory parameter of **HDFS->NameNode**.

Adjust the configuration in the system.

Step 8 Check whether the memory is configured properly based on the number of files in **Step 6** and the NameNode Heap Memory parameters in **Step 7**.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

NOTE

The recommended mapping between the number of HDFS file objects (filesystem objects = files + blocks) and the JVM parameters configured for NameNode is as follows:

- If the number of file objects reaches 10,000,000, you are advised to set the JVM parameters as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
- If the number of file objects reaches 20,000,000, you are advised to set the JVM parameters as follows: -Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G
- If the number of file objects reaches 50,000,000, you are advised to set the JVM parameters as follows: -Xms32G -Xmx32G -XX:NewSize=3G -XX:MaxNewSize=3G
- If the number of file objects reaches 100,000,000, you are advised to set the JVM parameters as follows: -Xms64G -Xmx64G -XX:NewSize=6G -XX:MaxNewSize=6G
- If the number of file objects reaches 200,000,000, you are advised to set the JVM parameters as follows: -Xms96G -Xmx96G -XX:NewSize=9G -XX:MaxNewSize=9G
- If the number of file objects reaches 300,000,000, you are advised to set the JVM parameters as follows: -Xms164G -Xmx164G -XX:NewSize=12G -XX:MaxNewSize=12G

Step 9 Modify the heap memory parameters of the NameNode based on the mapping between the number of file objects and the memory. Click **Save** and choose **Dashboard** > **More** > **Restart Service**.

Step 10 Check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 12 Select the following nodes in the required cluster from the **Service**:

- ZooKeeper
- HDFS

Step 13 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.81 ALM-14008 DataNode Heap Memory Usage Exceeds the Threshold

Description

The system checks the HDFS DataNode Heap Memory usage every 30 seconds and compares the actual Heap Memory usage with the threshold. The HDFS DataNode Heap Memory usage has a default threshold. This alarm is generated when the HDFS DataNode Heap Memory usage exceeds the threshold.

You can change the threshold in **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* **> HDFS**.

When the **Trigger Count** is 1, this alarm is cleared when the HDFS DataNode Heap Memory usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the HDFS DataNode Heap Memory usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The HDFS DataNode Heap Memory usage is too high, which affects the data read/write performance of the HDFS.

Possible Causes

The HDFS DataNode Heap Memory is insufficient.

Procedure

Delete unnecessary files.

Step 1 Log in to the HDFS client as user **root**. Run **cd** to switch to the client installation directory, and run **source bigdata_env**.

If the cluster uses the security mode, perform security authentication.

Run the **kinit hdfs** command and enter the password as prompted. Obtain the password from the administrator.

Step 2 Run the **hdfs dfs -rm -r** *file or directory* command to delete unnecessary files.

Step 3 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check the DataNode JVM memory usage and configuration.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**.

Step 5 In the **Basic Information** area, click **NameNode(Active)** to go to the HDFS WebUI.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 6 On the HDFS WebUI, click the **DataNodes** tab, and check the number of blocks of all DataNodes related to the alarm.

Step 7 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations** > **All Configurations**. In **Search**, enter **GC_OPTS** to check the GC_OPTS memory parameter of **HDFS->DataNode**.

Adjust the configuration in the system.

Step 8 Check whether the memory is configured properly based on the number of block in **Step 6** and the DataNode Heap Memory parameters in **Step 7**.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

NOTE

The mapping between the average number of blocks of a DataNode instance and the DataNode memory is as follows:

- If the average number of blocks of a DataNode instance reaches 2,000,000, the reference values of the JVM parameters of the DataNode are as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
- If the average number of blocks of a DataNode instance reaches 5,000,000, the reference values of the JVM parameters of the DataNode are as follows: -Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G

Step 9 Modify the heap memory parameters of the DataNode based on the mapping between the number of blocks and the memory. Click **Save** and choose **Dashboard** > **More** > **Restart Service**.

Step 10 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 12 Select **HDFS** in the required cluster from the **Service**.

Step 13 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.82 ALM-14009 Number of Dead DataNodes Exceeds the Threshold

Description

The system periodically detects the number of dead DataNodes in the HDFS cluster every 30 seconds, and compares the number with the threshold. The number of DataNodes in the Dead state has a default threshold. This alarm is generated when the number exceeds the threshold.

You can change the threshold in **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* **> HDFS**.

When the **Trigger Count** is 1, this alarm is cleared when the number of Dead DataNodes is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the number of Dead DataNodes is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

DataNodes that are in the Dead state cannot provide HDFS services.

Possible Causes

- DataNodes are faulty or overloaded.
- The network between the NameNode and the DataNode is disconnected or busy.
- NameNodes are overloaded.
- The NameNodes are not restarted after the DataNode is deleted.

Procedure

Check whether DataNodes are faulty.

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**. The **HDFS Status** page is displayed.

Step 2 In the **Basic Information** area, click **NameNode(Active)** to go to the HDFS WebUI.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 3 On the HDFS WebUI, click the **Datanodes** tab. In the **In operation** area, click **Filter** to check whether **down** is in the drop-down list.

- If yes, select **down**, record the information about the filtered DataNodes, and go to **Step 4**.
- If no, go to **Step 8**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Instance** to check whether recorded DataNodes exist in the instance list.

- If all recorded DataNodes exist, go to **Step 5**.
- If none of the recorded DataNodes exists, go to **Step 6**.
- If some of the recorded DataNodes exist, go to **Step 7**.

Step 5 Locate the DataNode instance, click **More** > **Restart Instance** to restart it and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Step 6 Select all NameNode instances, choose **More** > **Instance Rolling Restart** to restart them and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 16**.

Step 7 Select all NameNode instances, choose **More** > **Instance Rolling Restart** to restart them. Locate the DataNode instance, click **More** > **Restart Instance** to restart it and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check the status of the network between the NameNode and the DataNode.

Step 8 Log in to the faulty DataNode on the management page as user **root**, and run the **ping** *IP address of the NameNode* command to check whether the network between the DataNode and the NameNode is abnormal.

On the FusionInsight Manager page, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance**. In the instance list, view the service plane IP address of the faulty DataNode.

- If yes, go to **Step 9**.
- If no, go to **Step 10**.

Step 9 Rectify the network fault, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

Check whether the DataNode is overloaded.

Step 10 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and check whether the alarm **ALM-14008 HDFS DataNode Memory Usage Exceeds the Threshold** exists.

- If yes, go to **Step 11**.
- If no, go to **Step 13**.

Step 11 See **ALM-14008 HDFS DataNode Memory Usage Exceeds the Threshold** to handle the alarm and check whether the alarm is cleared.

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 Check whether the alarm is cleared from the alarm list.

- If yes, no further action is required.
- If no, go to **Step 13**.

Check whether the NameNode is overloaded.

Step 13 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and check whether the alarm **ALM-14007 HDFS NameNode Memory Usage Exceeds the Threshold** exists.

- If yes, go to **Step 14**.
- If no, go to **Step 16**.

Step 14 See **ALM-14007 HDFS NameNode Memory Usage Exceeds the Threshold** to handle the alarm and check whether the alarm is cleared.

- If yes, go to **Step 15**.
- If no, go to **Step 16**.

Step 15 Check whether the alarm is cleared from the alarm list.

- If yes, no further action is required.
- If no, go to **Step 16**.

Collect fault information.

Step 16 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 17 Select **HDFS** in the required cluster from the **Service**.

Step 18 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 19 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.83 ALM-14010 NameService Service Is Abnormal

Description

The system checks the NameService service status every 180 seconds. This alarm is generated when the NameService service is unavailable.

This alarm is cleared when the NameService service recovers.

Attribute

Alarm ID	Alarm Severity	Auto Clear
14010	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.

Impact on the System

HDFS fails to provide services for upper-layer components based on the NameService service, such as HBase and MapReduce. As a result, users cannot read or write files.

Possible Causes

- The KrbServer service is abnormal.
- The JournalNode is faulty.
- The DataNode is faulty.
- The disk capacity is insufficient.
- The NameNode enters safe mode.

Procedure

Check the KrbServer service status.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 2 Check whether the KrbServer service exists.

- If yes, go to **Step 3**.
- If no, go to **Step 6**.

Step 3 Click **KrbServer**.

Step 4 Click **Instances**. On the KrbServer management page, select the faulty instance, and choose **More** > **Restart Instance**. Check whether the instance successfully restarts.

- If yes, go to **Step 5**.
- If no, go to **Step 24**.

Step 5 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check the JournalNode instance status.

Step 6 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 7 Choose **HDFS** > **Instances**.

Step 8 Check whether the **Running Status** of the JournalNode is **Normal**.

- If yes, go to **Step 11**.
- If no, go to **Step 9**.

Step 9 Select the faulty JournalNode, and choose **More** > **Restart Instance**. Check whether the JournalNode successfully restarts.

- If yes, go to **Step 10**.
- If no, go to **Step 24**.

Step 10 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Check the DataNode instance status.

Step 11 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**.

Step 12 Click **Instances** and check whether **Running Status** of all DataNodes is **Normal**.

- If yes, go to **Step 15**.
- If no, go to **Step 13**.

Step 13 Click **Instances**. On the DataNode management page, select the faulty instance, and choose **More** > **Restart Instance**. Check whether the DataNode successfully restarts.

- If yes, go to **Step 14**.
- If no, go to **Step 15**.

Step 14 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Check disk status.

Step 15 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Host**.

Step 16 In the **Disk** column, check whether the disk space is insufficient.

- If yes, go to **Step 17**.
- If no, go to **Step 19**.

Step 17 Expand the disk capacity.

Step 18 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 19**.

Check whether NameNode is in the safe mode.

Step 19 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**. Click **NameNode(Active)** of the abnormal NameService. The NameNode web UI is displayed.

NOTE

By default, the admin user does not have the management rights of other components. If the page cannot be opened or the content is not completely displayed due to insufficient permission when you access the native page of a component, you can manually create a user with the management rights of the corresponding component to log in to the component.

Step 20 On the NameNode web UI, check whether "Safe mode is ON." is displayed.

Information behind **Safe mode is ON** is alarm information and is displayed based actual conditions.

- If yes, go to **Step 21**.
- If no, go to **Step 24**.

Step 21 Log in to the client as user **root**. Run the **cd** command to go to the client installation directory and run the **source bigdata_env** command. If the cluster uses the security mode, perform security authentication. Run the **kinit hdfs** command and enter the password as prompted. The password can be obtained from the administrator. If the cluster uses the non-security mode, log in as user **omm** and run the command. Ensure that user **omm** has the client execution permission.

Step 22 Run **hdfs dfsadmin -safemode leave**.

Step 23 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 24**.

Collect the fault information.

Step 24 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 25 In the **Service** area, select the following nodes of the desired cluster.

- ZooKeeper
- HDFS

Step 26 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 27 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.84 ALM-14011 DataNode Data Directory Is Not Configured Properly

Description

The DataNode parameter **dfs.datanode.data.dir** specifies DataNode data directories. This alarm is generated when a configured data directory cannot be created, a data directory uses the same disk as other critical directories in the system, or multiple directories use the same disk immediately.

This alarm is cleared when the DataNode data directory is configured properly and this DataNode for which the alarm is generated is restarted.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14011	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

If the DataNode data directory is mounted to the root directory or a critical directory, the disk space of the root directory or critical directory will be used up after long time running and the system will be faulty.

If the DataNode data directory is not configured properly, HDFS performance will deteriorate.

Possible Causes

- The DataNode data directory fails to be created.
- The DataNode data directory uses the same disk with critical directories, such as **/** or **/boot**.

- Multiple directories in the DataNode data directory use the same disk.

Procedure

Check the alarm cause and information about the DataNode for which the alarm is generated.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. In the alarm list, click the alarm.

Step 2 In **HostName** of **Location**, obtain the host name of the DataNode for which the alarm is generated.

Delete directories that do not comply with the disk plan from the DataNode data directory.

Step 3 Choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance**. In the instance list, click the DataNode instance on the node for which the alarm is generated.

Step 4 Click **Instance Configurations** and view the value of the DataNode parameter **dfs.datanode.data.dir**.

Step 5 Check whether all DataNode data directories are consistent with the disk plan.

- If yes, go to **Step 6**.
- If no, go to **Step 9**.

Step 6 Modify the DataNode parameter **dfs.datanode.data.dir** and delete the incorrect directories.

Step 7 Choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** and restart the DataNode instance.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Step 9 Log in to the DataNode for which the alarm is generated as user **root**.

- If the alarm cause is "The DataNode data directory fails to be created", go to **Step 10**.
- If the alarm cause is "The DataNode data directory uses the same disk with critical directories, such **/** or **/boot**", go to **Step 17**.
- If the alarm cause is "Multiple directories in the DataNode data directory uses the same disk", go to **Step 21**.

Check whether the DataNode data directory fails to be created.

Step 10 Run the **su - omm** command to switch to user **omm**.

Step 11 Run the **ls** command to check whether the directories exist in the DataNode data directory.

- If yes, go to **Step 26**.
- If no, go to **Step 12**.

Step 12 Run the **mkdir** *data directory* command to create the directory and check whether the directory can be successfully created.

- If yes, go to **Step 24**.
- If no, go to **Step 13**.

Step 13 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** to check whether alarm **ALM-12017 Insufficient Disk Capacity** exists.

- If yes, go to **Step 14**.
- If no, go to **Step 15**.

Step 14 Adjust the disk capacity and check whether alarm **ALM-12017 Insufficient Disk Capacity** is cleared. For details, see **ALM-12017 Insufficient Disk Capacity**.

- If yes, go to **Step 12**.
- If no, go to **Step 15**.

Step 15 Check whether user **omm** has the **rwx** or **x** permission of all the upper-layer directories of the directory. (For example, for **/tmp/abc/**, user **omm** has the **x** permission for directory **tmp** and the **rwx** permission for directory **abc**.)

- If yes, go to **Step 24**.
- If no, go to **Step 16**.

Step 16 Run the **chmod u+rwx** *path* or **chmod u+x** *path* command as user **root** to assign the **rwx** or **x** permission of these directories to user **omm**. Then go to **Step 12**.

Check whether the DataNode data directory use the same disk as other critical directories in the system.

Step 17 Run the **df** command to obtain the disk mounting information of each directory in the DataNode data directory.

Step 18 Check whether the directories mounted to the disk are critical directories, such as **/** or **/boot**.

- If yes, go to **Step 19**.
- If no, go to **Step 24**.

Step 19 Change the value of the DataNode parameter **dfs.datanode.data.dir** and delete the directories that use the same disk as critical directories.

Step 20 Go to **Step 24**.

Check whether multiple directories in the DataNode data directory use the same disk.

Step 21 Run the **df** command to obtain the disk mounting information of each directory in the DataNode data directory. Record the mounted directory in the command output.

Step 22 Modify the DataNode node parameters **dfs.datanode.data.dir** to reserve only one directory among the directories that mounted to the same disk directory.

Step 23 Go to **Step 24**.

Restart the DataNode and check whether the alarm is cleared.

Step 24 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** and restart the DataNode instance

Step 25 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 26**.

Collect fault information.

Step 26 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 27 Select **HDFS** in the required cluster from the **Service**.

Step 28 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 29 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.85 ALM-14012 JournalNode Is Out of Synchronization

Description

On the active NameNode, the system checks the data consistency of all JournalNodes in the cluster every 5 minutes. This alarm is generated when the data on a JournalNode is inconsistent with the data on the other JournalNodes.

This alarm is cleared in 5 minutes after the data on JournalNodes is consistent.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14012	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.

Impact on the System

When a JournalNode is working incorrectly, the data on the node becomes inconsistent with that on the other JournalNodes. If data on more than half of JournalNodes is inconsistent, the NameNode cannot work correctly, making the HDFS service unavailable.

Possible Causes

- The JournalNode instance does not exist (deleted or migrated).
- The JournalNode instance has not been started or has been stopped.
- The JournalNode instance is working incorrectly.
- The network of the JournalNode is unreachable.

Procedure

Check whether the JournalNode instance has been started up.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. In the alarm list, click the alarm.

Step 2 Check **Location** and obtain the IP address of the JournalNode for which the alarm is generated.

Step 3 Choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance**. In the instance list, check whether the JournalNode instance exists on the node for which the alarm is generated.

- If yes, go to **Step 5**.
- If no, go to **Step 4**.

Step 4 Choose **O&M** > **Alarm** > **Alarms**. In the alarm list, click **Clear** in the **Operation** column of the alarm. In the dialog box that is displayed, click **OK**. No further action is needed.

Step 5 Click the JournalNode instance and check whether its **Configuration Status** is **Synchronized**.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 Select the JournalNode instance and choose **Start Instance** to start the instance.

Step 7 After 5 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Check whether the JournalNode instance is working correctly.

Step 8 Check whether **Running Status** of the JournalNode instance is **Normal**.

- If yes, go to **Step 11**.
- If no, go to **Step 9**.

Step 9 Select the JournalNode instance and choose **More** > **Restart Instance** to start the instance.

Step 10 After 5 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Check whether the network of the JournalNode is reachable.

Step 11 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** to check the service IP address of the active NameNode.

Step 12 Log in to the active NameNode as user **root**.

Step 13 Run the **ping** command to check whether a timeout occurs or the network is unreachable between the active NameNode and the JournalNode.

ping *service IP address of the JournalNode*

- If yes, go to **Step 14**.
- If no, go to **Step 15**.

Step 14 Contact the network administrator to rectify the network fault and check whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, go to **Step 15**.

Collect fault information.

Step 15 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 16 Select **HDFS** in the required cluster from the **Service**.

Step 17 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.86 ALM-14013 Failed to Update the NameNode FsImage File

Description

HDFS metadata is stored in the FsImage file of the NameNode data directory, which is specified by the **dfs.namenode.name.dir** configuration item. The standby NameNode periodically combines existing FsImage files and Editlog files stored in the JournalNode to generate a new FsImage file, and then pushes the new FsImage file to the data directory of the active NameNode. This period is specified by the **dfs.namenode.checkpoint.period** configuration item of HDFS. The default value is 3600s, namely, one hour. If the FsImage file in the data directory of the active NameNode is not updated, the HDFS metadata combination function is abnormal and requires rectification.

On the active NameNode, the system checks the FsImage file information every five minutes. This alarm is generated when no FsImage file is generated within three combination periods.

This alarm is cleared when a new FsImage file is generated and pushed to the active NameNode, which indicates that the HDFS metadata combination function can be properly used.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14013	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.

Impact on the System

If the FsImage file in the data directory of the active NameNode is not updated, the HDFS metadata combination function is abnormal and requires rectification. If it is not rectified, the Editlog files increase continuously after HDFS runs for a period. In this case, HDFS restart is time-consuming because a large number of Editlog files need to be loaded. In addition, this alarm also indicates that the standby NameNode is abnormal and the NameNode high availability (HA) mechanism becomes invalid. When the active NameNode is faulty, the HDFS service becomes unavailable.

Possible Causes

- The standby NameNode is stopped.
- The standby NameNode instance is working incorrectly.
- The standby NameNode fails to generate a new FsImage file.
- Space of the data directory on the standby NameNode is insufficient.
- The standby NameNode fails to push the FsImage file to the active NameNode.
- Space of the data directory on the active NameNode is insufficient.

Procedure

Check whether the standby NameNode is stopped.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. In the alarm list, click the alarm.

Step 2 View **Location** and obtain the host name of the active NameNode for which the alarm is generated and name of the NameService where the active NameNode resides.

Step 3 Choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance**, find the standby NameNode instance of the NameService in the instance list, and check whether its **Configuration Status** is **Synchronized**.

- If yes, go to **Step 6**.
- If no, go to **Step 4**.

Step 4 Select the standby NameNode instance, choose **Start Instance**, and wait until the startup is complete.

Step 5 Wait for a NameNode metadata combination period and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check whether the NameNode instance is working correctly.

Step 6 Check whether **Running Status** of the standby NameNode instance is **Normal**.

- If yes, go to **Step 9**.
- If no, go to **Step 7**.

Step 7 Select the standby NameNode instance, choose **More** > **Restart Instance**, and wait until the startup is complete.

Step 8 Wait for a NameNode metadata combination period and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 30**.

Check whether the standby NameNode fails to generate a new FsImage file.

Step 9 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations**, and search and obtain the value of **dfs.namenode.checkpoint.period**. This value is the period of NameNode metadata combination.

Step 10 Choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** and obtain the service IP addresses of the active and standby NameNodes of the NameService for which the alarm is generated.

Step 11 Click the **NameNode(***xx***,Standy)** and **Instance Configurations** to obtain the value of **dfs.namenode.name.dir**. This value is the FsImage storage directory of the standby NameNode.

Step 12 Log in to the standby NameNode as user **root** or **omm**.

Step 13 Go to the FsImage storage directory and check the generation time of the newest FsImage file.

cd *Storage directory of the standby NameNode***/current**

stat -c %y $(ls -t | grep "fsimage_[0-9]*$" | head -1)

Step 14 Run the **date** command to obtain the current system time.

Step 15 Calculate the time difference between the generation time of the newest FsImage file and the current system time and check whether the time difference is greater than three times of the metadata combination period.

- If yes, go to **Step 16**.
- If no, go to **Step 20**.

Step 16 The metadata combination function of the standby NameNode is faulty. Run the following command to check whether the fault is caused by insufficient storage space.

Go to the FsImage storage directory and check the size of the newest FsImage file (in MB).

cd *Storage directory of the standby NameNode***/current**

du -m $(ls -t | grep "fsimage_[0-9]*$" | head -1) | awk '{print $1}'

Step 17 Run the following command to check the available disk space of the standby NameNode (in MB).

df -m ./ | awk 'END{print $4}'

Step 18 Compare the FsImage file size and the available disk space and determine whether another FsImage file can be stored on the disk.

- If yes, go to **Step 7**.
- If no, go to **Step 19**.

Step 19 Clear the redundant files on the disk where the directory resides to reserve sufficient space for metadata. After the clearance, wait for a NameNode metadata combination period and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 20**.

Check whether the standby NameNode fails to push the FsImage file to the active NameNode.

Step 20 Log in to the standby NameNode as user **root**.

Step 21 Run the **su - omm** command to switch to user **omm**.

Step 22 Run the following command to check whether the standby NameNode can push the file to the active NameNode.

tmpFile=/tmp/tmp_test_$(date +%s)

echo "test" > $tmpFile

scp $tmpFile *Service IP address of the active NameNode*:**/tmp**

- If yes, go to **Step 24**.
- If no, go to **Step 23**.

Step 23 When the standby NameNode fails to push data to the active NameNode as user **omm**, contact the system administrator to handle the fault. Wait for a NameNode metadata combination period and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 24**.

Check whether space on the data directory of the active NameNode is insufficient.

Step 24 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Instance**, click the active NameNode of the NameService for which the alarm is generated, and then click **Instance Configurations** to obtain the value of **dfs.namenode.name.dir**. This value is the FsImage storage directory of the active NameNode.

Step 25 Log in to the active NameNode as user **root** or **omm**.

Step 26 Go to the FsImage storage directory and check the size of the newest FsImage file (in MB).

cd *Storage directory of the active NameNode*/**current**

du -m $(ls -t | grep "fsimage_[0-9]*$" | head -1) | awk '{print $1}'

Step 27 Run the following command to check the available disk space of the active NameNode (in MB).

df -m ./ | awk 'END{print $4}'

Step 28 Compare the FsImage file size and the available disk space and determine whether another FsImage file can be stored on the disk.

- If yes, go to **Step 30**.
- If no, go to **Step 29**.

Step 29 Clear the redundant files on the disk where the directory resides to reserve sufficient space for metadata. After the clearance, wait for a NameNode metadata combination period and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 30**.

Collect fault information.

Step 30 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 31 Select **NameNode** in the required cluster from the **Service**.

Step 32 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 33 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.87 ALM-14014 NameNode GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the NameNode process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default).

This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14014	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the NameNode process may interrupt the services.

Possible Causes

The heap memory of the NameNode instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **ALM-14014 NameNode GC Time Exceeds the Threshold.** Then check the role name in **Location** and confirm the IP adress of the instance.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** > **NameNode (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Garbage Collection**, and select **NameNode Garbage Collection (GC)** to check the GC duration statistics of the NameNode process collected every minute.

Step 3 Check whether the GC duration of the NameNode process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations** > **NameNode** > **System** to increase the value of **GC_OPTS** parameter as required.

NOTE

The recommended mapping between the number of HDFS file objects (filesystem objects = files + blocks) and the JVM parameters configured for NameNode is as follows:

- If the number of file objects reaches 10,000,000, you are advised to set the JVM parameters as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
- If the number of file objects reaches 20,000,000, you are advised to set the JVM parameters as follows: -Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G
- If the number of file objects reaches 50,000,000, you are advised to set the JVM parameters as follows: -Xms32G -Xmx32G -XX:NewSize=3G -XX:MaxNewSize=3G
- If the number of file objects reaches 100,000,000, you are advised to set the JVM parameters as follows: -Xms64G -Xmx64G -XX:NewSize=6G -XX:MaxNewSize=6G
- If the number of file objects reaches 200,000,000, you are advised to set the JVM parameters as follows: -Xms96G -Xmx96G -XX:NewSize=9G -XX:MaxNewSize=9G
- If the number of file objects reaches 300,000,000, you are advised to set the JVM parameters as follows: -Xms164G -Xmx164G -XX:NewSize=12G -XX:MaxNewSize=12G

Step 5 Save the configuration and restart the NameNode instance.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 8 Select **NameNode** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.88 ALM-14015 DataNode GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the DataNode process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default).

This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14015	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the DataNode process may interrupt the services.

Possible Causes

The heap memory of the DataNode instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **ALM-14015 DataNode GC Time Exceeds the Threshold**. Then check the role name in **Location** and confirm the IP adress of the instance.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** > **DataNode (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Garbage Collection**, and select **DataNode Garbage Collection (GC)** to check the GC duration statistics of the DataNode process collected every minute.

Step 3 Check whether the GC duration of the DataNode process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations** > **DataNode** > **System** to increase the value of **GC_OPTS** parameter as required.

NOTE

The mapping between the average number of blocks of a DataNode instance and the DataNode memory is as follows:

- If the average number of blocks of a DataNode instance reaches 2,000,000, the reference values of the JVM parameters of the DataNode are as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
- If the average number of blocks of a DataNode instance reaches 5,000,000, the reference values of the JVM parameters of the DataNode are as follows: -Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G

Step 5 Save the configuration and restart the DataNode instance.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **DataNode** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.89 ALM-14016 DataNode Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the HDFS service every 30 seconds. This alarm is generated when the direct memory usage of a DataNode instance exceeds the threshold (90% of the maximum memory).

The alarm is cleared when the direct memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14016	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the HDFS service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the DataNode instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**. On the displayed interface, click the drop-down button of **ALM-14016 DataNode Direct Memory Usage Exceeds the Threshold**. Then check the role name in **Location** and confirm the IP adress of the instance.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** > **DataNode (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Resource**, and select **DataNode Memory** to check the direct memory usage.

Step 3 Check whether the used direct memory of DataNode reaches 90% of the maximum direct memory specified for DataNode by default.

- If yes, go to **Step 4**.
- If no, go to **Step 8**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations** > **DataNode** > **System** to check whether "-XX:MaxDirectMemorySize" exists in the **GC_OPTS** parameter.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 In the **GC_OPTS** parameter, delete "-XX:MaxDirectMemorySize". Save the configuration and restart the DataNode instance.

Step 6 Check whether the **ALM-14008 DataNode Heap Memory Usage Exceeds the Threshold** exists.

- If yes, handle the alarm by referring to **ALM-14008 DataNode Heap Memory Usage Exceeds the Threshold**.
- If no, go to **Step 7**.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 9 Select **DataNode** in the required cluster from the **Service**.

Step 10 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.90 ALM-14017 NameNode Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the HDFS service every 30 seconds. This alarm is generated when the direct memory usage of a NameNode instance exceeds the threshold (90% of the maximum memory).

The alarm is cleared when the direct memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14017	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the HDFS service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the NameNode instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms.** On the displayed interface, click the drop-down button of **ALM-14017 NameNode Direct Memory Usage Exceeds the Threshold**. Then check the role name in **Location** and confirm the IP adress of the instance.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Instance** > **NameNode (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Resource**, and select **NameNode Memory** to check the direct memory usage.

Step 3 Check whether the used direct memory of NameNode reaches 90% of the maximum direct memory specified for NameNode by default.

- If yes, go to **Step 4**.
- If no, go to **Step 8**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations** > **NameNode** > **System** to check whether "-XX:MaxDirectMemorySize" exists in the **GC_OPTS** parameter.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 In the **GC_OPTS** parameter, delete "-XX:MaxDirectMemorySize". Save the configuration and restart the NameNode instance.

Step 6 Check whether the **ALM-14007 NameNode Heap Memory Usage Exceeds the Threshold** exists.

- If yes, handle the alarm by referring to **ALM-14007 NameNode Heap Memory Usage Exceeds the Threshold**.
- If no, go to **Step 7**.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 9 Select **NameNode** in the required cluster from the **Service**.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.91 ALM-14018 NameNode Non-heap Memory Usage Exceeds the Threshold

Description

The system checks the non-heap memory usage of the HDFS NameNode every 30 seconds and compares the actual usage with the threshold. The non-heap memory usage of the HDFS NameNode has a default threshold. This alarm is generated when the non-heap memory usage of the HDFS NameNode exceeds the threshold.

Users can choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* **> HDFS** to change the threshold.

This alarm is cleared when the no-heap memory usage of the HDFS NameNode is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14018	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the memory usage of the HDFS NameNode is too high, data read/write performance of HDFS will be affected.

Possible Causes

Non-heap memory of the HDFS NameNode is insufficient.

Procedure

Delete unnecessary files.

Step 1 Log in to the HDFS client as user **root**. Run the **cd** command to go to the client installation directory, and run the **source bigdata_env** command.

If the cluster adopts the security mode, perform security authentication.

Run the **kinit hdfs** command and enter the password as prompted. Obtain the password from the administrator.

Step 2 Run the **hdfs dfs -rm -r** *file or directory path* command to delete unnecessary files.

Step 3 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check the NameNode JVM non-heap memory usage and configuration.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS**. The HDFS status page is displayed.

Step 5 In the **Basic Information** area, click **NameNode(Active)**. The HDFS WebUI is displayed.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 6 On the HDFS WebUI, click the **Overview** tab. In **Summary**, check the numbers of files, directories, and blocks in HDFS.

Step 7 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations**. In **Search**,

enter **GC_OPTS** to check the **GC_OPTS** non-heap memory parameter of **HDFS->NameNode**.

Adjust system configurations.

Step 8 Check whether the non-heap memory is properly configured based on the number of file objects in **Step 6** and the non-heap parameters configured for NameNode in **Step 7**.

- If yes, go to **Step 9**.
- If no, go to **Step 12**.

NOTE

The recommended mapping between the number of HDFS file objects (filesystem objects = files + blocks) and the JVM parameters configured for NameNode is as follows:

- If the number of file objects reaches 10,000,000, you are advised to set the JVM parameters as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
- If the number of file objects reaches 20,000,000, you are advised to set the JVM parameters as follows: -Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G
- If the number of file objects reaches 50,000,000, you are advised to set the JVM parameters as follows: -Xms32G -Xmx32G -XX:NewSize=3G -XX:MaxNewSize=3G
- If the number of file objects reaches 100,000,000, you are advised to set the JVM parameters as follows: -Xms64G -Xmx64G -XX:NewSize=6G -XX:MaxNewSize=6G
- If the number of file objects reaches 200,000,000, you are advised to set the JVM parameters as follows: -Xms96G -Xmx96G -XX:NewSize=9G -XX:MaxNewSize=9G
- If the number of file objects reaches 300,000,000, you are advised to set the JVM parameters as follows: -Xms164G -Xmx164G -XX:NewSize=12G -XX:MaxNewSize=12G

Step 9 Modify the **GC_OPTS** parameter of the NameNode based on the mapping between the number of file objects and non-heap memory.

Step 10 Save the configuration and click **Dashboard** > **More** > **Restart Service**.

Step 11 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Collect fault information.

Step 12 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 13 Select the following services in the required cluster from the **Service**.

- ZooKeeper
- HDFS

Step 14 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 15 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.92 ALM-14019 DataNode Non-heap Memory Usage Exceeds the Threshold

Description

The system checks the non-heap memory usage of the HDFS DataNode every 30 seconds and compares the actual usage with the threshold. The non-heap memory usage of the HDFS DataNode has a default threshold. This alarm is generated when the non-heap memory usage of the HDFS DataNode exceeds the threshold.

Users can choose **O&M > Alarm > Thresholds>** *Name of the desired cluster* **> HDFS** to change the threshold.

This alarm is cleared when the no-heap memory usage of the HDFS DataNode is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14019	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the memory usage of the HDFS DataNode is too high, data read/write performance of HDFS will be affected.

Possible Causes

Non-heap memory of the HDFS DataNode is insufficient.

Procedure

Delete unnecessary files.

Step 1 Log in to the HDFS client as user **root**. Run the **cd** command to go to the client installation directory, and run the **source bigdata_env** command.

If the cluster adopts the security mode, perform security authentication.

Run the **kinit hdfs** command and enter the password as prompted. Obtain the password from the administrator.

Step 2 Run the **hdfs dfs -rm -r** *file or directory path* command to delete unnecessary files.

Step 3 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check the DataNode JVM non-heap memory usage and configuration.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**.

Step 5 In the **Basic Information** area, click **NameNode(Active)**. The HDFS WebUI is displayed.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 6 On the HDFS WebUI, click the **Datanodes** tab to view the number of blocks of all DataNodes that report alarms.

Step 7 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations** > **All Configurations**. In **Search**, enter **GC_OPTS** to check the **GC_OPTS** non-heap memory parameter of **HDFS->DataNode**.

Adjust system configurations.

Step 8 Check whether the memory is properly configured based on the number of blocks in **Step 6** and the memory parameters configured for DataNode in **Step 7**.

- If yes, go to **Step 9**.
- If no, go to **Step 12**.

NOTE

The mapping between the average number of blocks of a DataNode instance and the DataNode memory is as follows:

- If the average number of blocks of a DataNode instance reaches 2,000,000, the reference values of the JVM parameters of the DataNode are as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
- If the average number of blocks of a DataNode instance reaches 5,000,000, the reference values of the JVM parameters of the DataNode are as follows: -Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G

Step 9 Modify the **GC_OPTS** parameter of the DataNode based on the mapping between the number of blocks and memory.

Step 10 Save the configuration and click **Dashboard** > **More** > **Restart Service**.

Step 11 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Collect fault information.

Step 12 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 13 Select the following services in the required cluster from the **Service**.

- ZooKeeper
- HDFS

Step 14 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 15 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.93 ALM-14020 Number of Entries in the HDFS Directory Exceeds the Threshold

Description

The system obtains the number of subfiles and subdirectories in a specified directory every hour and checks whether it reaches the percentage of the threshold (the maximum number of subfiles and subdirectories in an HDFS directory, the threshold for triggering an alarm is **90%** by default). If it exceeds the percentage of the threshold, an alarm is triggered.

When the number of subfiles and subdirectories in the directory the alarm is lower than the percentage of the threshold, the alarm is automatically cleared. When the monitoring switch is disabled, alarms corresponding to all directories are cleared. If a directory is removed from the monitoring list, alarms corresponding to the directory are cleared.

NOTE

- The **dfs.namenode.fs-limits.max-directory-items** parameter specifies the maximum number of subfiles and subdirectories in the HDFS directory. Its default value is **1048576**. If the number of subfiles and subdirectories in a directory exceeds the parameter value, subfiles and subdirectories cannot be created in the directory.
- The **dfs.namenode.directory-items.monitor** parameter specifies the list of directories to be monitored. Its default value is **/tmp,/SparkJobHistory,/mr-history**.
- The **dfs.namenode.directory-items.monitor.enabled** parameter is used to enable or disable the monitoring switch. Its default value is **true**, which means the monitoring switch is enabled by default.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14020	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
NameServiceName	Specifies the NameService service for which the alarm is generated.
Directory	Specifies the directory for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the number of entries in the monitored directory exceeds 90% of the threshold, an alarm is triggered, but entries can be added to the directory. Once the maximum threshold is exceeded, entries will fail to be added to the directory.

Possible Causes

The number of entries in the monitored directory exceeds 90% of the threshold.

Procedure

Check whether unnecessary files exist in the system.

Step 1 Log in to the HDFS client as user **root**. Run the **cd** command to go to the client installation directory, and run the **source bigdata_env** command to set the environment variables.

If the cluster is in security mode, security authentication is required.

Run the **kinit hdfs** command and enter the password as prompted. Obtain the password from the administrator.

Step 2 Run the following command to check whether files and directories in the directory with the alarm can be deleted:

hdfs dfs -ls *Directory with the alarm*

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Run the following command to delete unnecessary files.

hdfs dfs -rm -r -f *File or directory path*

NOTE

Deleting a file or folder is a high-risk operation. Ensure that the file or folder is no longer required before performing this operation.

Step 4 Wait 1 hour and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the threshold is correctly configured.

Step 5 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations**. Search for the **dfs.namenode.fs-limits.max-directory-items** parameter and check whether the parameter value is appropriate.

- If yes, go to **Step 9**.
- If no, go to **Step 6**.

Step 6 Increase the parameter value.

Step 7 Save the configuration and click **Dashboard** > **More** > **Restart Service**.

Step 8 Wait 1 hour and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **HDFS** in the required cluster from the **Service**.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.94 ALM-14021 NameNode Average RPC Processing Time Exceeds the Threshold

Description

The system checks the average RPC processing time of NameNode every 30 seconds, and compares the actual average RPC processing time with the threshold (default value: 100 ms). This alarm is generated when the system detects that the average RPC processing time exceeds the threshold for several consecutive times (10 times by default).

You can choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **HDFS** to change the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the average RPC processing time of NameNode is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the average RPC processing time of NameNode is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14021	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService service for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

NameNode cannot process the RPC requests from HDFS clients, upper-layer services that depend on HDFS, and DataNode in a timely manner. Specifically, the services that access HDFS run slowly or the HDFS service is unavailable.

Possible Causes

- The CPU performance of NameNode nodes is insufficient and therefore NameNode nodes cannot process messages in a timely manner.
- The configured NameNode memory is too small and frame freezing occurs on the JVM due to frequent full garbage collection.
- NameNode parameters are not configured properly, so NameNode cannot make full use of system performance.

Procedure

Obtain alarm information.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. In the alarm list, click the alarm.

Step 2 Check the alarm. Obtain the host name of the NameNode node involved in this alarm from the **HostName** information of **Location**. Then obtain the name of the NameService node involved in this alarm from the **NameServiceName** information of **Location**.

Check whether the threshold is too small.

Step 3 Check the status of the services that depend on HDFS. Check whether the services run slowly or task execution times out.

- If yes, go to **Step 8**.
- If no, go to **Step 4**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **RPC**, and select **Average Time of Active NameNode RPC Processing** and click **OK**.

Step 5 On the **Average Time of Active NameNode RPC Processing** monitoring page, obtain the value of the NameService node involved in this alarm.

Step 6 On the FusionInsight Manager portal, choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* **> HDFS**. Locate **Average Time of Active NameNode RPC Processing** and click the **Modify** in the **Operation** column of the default rule. The **Modify Rule** page is displayed. Change **Threshold** to 150% of the peak value within one day before and after the alarm is generated. Click **OK** to save the new threshold.

Step 7 Wait for 5 minutes and then check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check whether the CPU performance of the NameNode node is sufficient.

Step 8 On the FusionInsight Manager portal, click **O&M > Alarm >Alarms** and check whether **ALM-12016 CPU Usage Exceeds the Threshold** is generated for the NameNode node.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

Step 9 Handle **ALM-12016 CPU Usage Exceeds the Threshold** by taking recommended actions.

Step 10 Wait for 10 minutes and check whether alarm 14021 is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Check whether the memory of the NameNode node is too small.

Step 11 On the FusionInsight Manager portal, click **O&M > Alarm >Alarms** and check whether **ALM-14007 HDFS NameNode Heap Memory Usage Exceeds the Threshold** is generated for the NameNode node.

- If yes, go to **Step 12**.
- If no, go to **Step 14**.

Step 12 Handle **ALM-14007 HDFS NameNode Heap Memory Usage Exceeds the Threshold** by taking recommended actions.

Step 13 Wait for 10 minutes and check whether alarm 14021 is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Check whether NameNode parameters are configured properly.

Step 14 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations**. Search for parameter **dfs.namenode.handler.count** and view its value. If the value is less

than or equal to 128, change it to **128**. If the value is greater than 128 but less than 192, change it to **192**.

Step 15 Search for parameter **ipc.server.read.threadpool.size** and view its value. If the value is less than 5, change it to **5**.

Step 16 Click **Save** and click **OK**.

Step 17 On the **Instance** page of HDFS, select the standby NameNode of NameService involved in this alarm and choose **More** > **Restart Instance**. Enter the password and click **OK**. Wait until the standby NameNode is started up.

Step 18 On the **Instance** page of HDFS, select the active NameNode of NameService involved in this alarm and choose **More** > **Restart Instance**. Enter the password and click **OK**. Wait until the active NameNode is started up.

Step 19 Wait for 1 hour and then check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 20**.

Collect fault information.

Step 20 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 21 Select the following node in the required cluster from the **Service**.

- HDFS

Step 22 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 23 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.95 ALM-14022 NameNode Average RPC Queuing Time Exceeds the Threshold

Description

The system checks the average RPC queuing time of NameNode every 30 seconds, and compares the actual average RPC queuing time with the threshold (default value: 200 ms). This alarm is generated when the system detects that the average RPC queuing time exceeds the threshold for several consecutive times (10 times by default).

You can choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **HDFS** to change the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the average RPC queuing time of NameNode is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the average RPC queuing time of NameNode is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14022	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService service for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

NameNode cannot process the RPC requests from HDFS clients, upper-layer services that depend on HDFS, and DataNode in a timely manner. Specifically, the services that access HDFS run slowly or the HDFS service is unavailable.

Possible Causes

- The CPU performance of NameNode nodes is insufficient and therefore NameNode nodes cannot process messages in a timely manner.
- The configured NameNode memory is too small and frame freezing occurs on the JVM due to frequent full garbage collection.
- NameNode parameters are not configured properly, so NameNode cannot make full use of system performance.
- The volume of services that access HDFS is too large and therefore NameNode is overloaded.

Procedure

Obtain alarm information.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. In the alarm list, click the alarm.

Step 2 Check the alarm. Obtain the alarm generation time from **Generated**. Obtain the host name of the NameNode node involved in this alarm from the **HostName** information of **Location**. Then obtain the name of the NameService node involved in this alarm from the **NameServiceName** information of **Location**.

Check whether the threshold is too small.

Step 3 Check the status of the services that depend on HDFS. Check whether the services run slowly or task execution times out.

- If yes, go to **Step 8**.
- If no, go to **Step 4**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **RPC**, and select **Average Time of Active NameNode RPC Queuing** and click **OK**.

Step 5 On the **Average Time of Active NameNode RPC Queuing** monitoring page, obtain the value of the NameService node involved in this alarm.

Step 6 On the FusionInsight Manager portal, choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* **> HDFS**. Locate **Average Time of Active NameNode RPC Queuing** and click the **Modify** in the **Operation** column of the default rule. The **Modify Rule** page is displayed. Change **Threshold** to 150% of the monitored value. Click **OK** to save the new threshold.

Step 7 Wait for 1 minute and then check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check whether the CPU performance of the NameNode node is sufficient.

Step 8 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms** and check whether **ALM-12016 HDFS NameNode Memory Usage Exceeds the Threshold** is generated.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

Step 9 Handle **ALM-12016 CPU Usage Exceeds the Threshold** by taking recommended actions.

Step 10 Wait for 10 minutes and check whether alarm 14022 is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Check whether the memory of the NameNode node is too small.

Step 11 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms** and check whether **ALM-14007 HDFS NameNode Memory Usage Exceeds the Threshold** is generated.

- If yes, go to **Step 12**.
- If no, go to **Step 14**.

Step 12 Handle **ALM-14007 CPU Usage Exceeds the Threshold** by taking recommended actions.

Step 13 Wait for 10 minutes and check whether alarm 14022 is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Check whether NameNode parameters are configured properly.

Step 14 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS** > **Configurations** > **All Configurations**. Search for parameter **dfs.namenode.handler.count** and view its value. If the value is less than or equal to 128, change it to **128**. If the value is greater than 128 but less than 192, change it to **192**.

Step 15 Search for parameter **ipc.server.read.threadpool.size** and view its value. If the value is less than 5, change it to **5**.

Step 16 Click **Save**, and click **OK**.

Step 17 On the **Instance** page of HDFS, select the standby NameNode of NameService involved in this alarm and choose **More** > **Restart Instance**. Enter the password and click **OK**. Wait until the standby NameNode is started up.

Step 18 On the **Instance** page of HDFS, select the active NameNode of NameService involved in this alarm and choose **More** > **Restart Instance**. Enter the password and click **OK**. Wait until the active NameNode is started up.

Step 19 Wait for 1 hour and then check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 20**.

Check whether the HDFS workload changes and reduce the workload properly.

Step 20 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **HDFS**. Click the drop-down menu in the upper right corner of **Chart**, click **Customize**, select **Average Time of Active NameNode RPC Queuing** and click **OK**.

Step 21 Click  . The **Details** page is displayed.

Step 22 Set the monitoring data display period, from 5 days before the alarm generation time to the alarm generation time. Click **OK**.

Step 23 On the **Average RPC Queuing Time** monitoring page, check whether the point in time when the queuing time increases abruptly exists.

- If yes, go to **Step 24**.
- If no, go to **Step 27**.

Step 24 Confirm and check the point in time. Check whether a new task frequently accesses HDFS and whether the access frequency can be reduced.

Step 25 If a Balancer task starts at the point in time, stop the task or specify a node for the task to reduce the HDFS workload.

Step 26 Wait for 1 hour and then check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 27**.

Collect fault information.

Step 27 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 28 Select **HDFS** in the required cluster from the **Service**.

Step 29 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 30 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.96 ALM-14023 Percentage of Total Reserved Disk Space for Replicas Exceeds the Threshold

Description

The system checks the percentage of total reserved disk space for replicas (Total reserved disk space for replicas/(Total reserved disk space for replicas + Total remaining disk space)) every 30 seconds and compares the actual percentage with the threshold (**90%** by default). This alarm is generated when the percentage of total reserved disk space for replicas exceeds the threshold for multiple consecutive times (**Trigger Count**).

The alarm is cleared in the following two scenarios: The value of **Trigger Count** is **1** and the percentage of total reserved disk space for replicas is less than or equal to the threshold; the value of **Trigger Count** is greater than **1** and the percentage of total reserved disk space for replicas is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14023	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
NameServiceName	Specifies the NameService service for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The performance of writing data to HDFS is affected. If all remaining DataNode space is reserved for replicas, writing HDFS data fails.

Possible Causes

- The alarm threshold is improperly configured.
- The disk space configured for the HDFS cluster is insufficient.
- The volume of services that access HDFS is too large and therefore DataNode is overloaded.

Procedure

Check whether the alarm threshold is appropriate.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **HDFS** > **Disk** > **Percentage of Reserved Space for Replicas of Unused Space** to check whether the alarm threshold is appropriate. (The default threshold is **90%**. Users can change it as required.)

- If yes, go to **Step 4**.
- If no, go to **Step 2**.

Step 2 Choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **HDFS** > **Disk** > **Percentage of Reserved Space for Replicas of Unused Space** and Click **Modify,** change the threshold based on the actual usage.

Step 3 Wait 5 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether an alarm indicating insufficient disk space is generated.

Step 4 On the FusionInsight Manager portal, check whether **ALM-14001 HDFS Disk Usage Exceeds the Threshold** or **ALM-14002 DataNode Disk Usage Exceeds the Threshold** exists on the **O&M > Alarm > Alarms** page.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 Handle the alarm by referring to instructions in **ALM-14001 HDFS Disk Usage Exceeds the Threshold** or **ALM-14002 DataNode Disk Usage Exceeds the Threshold** and check whether the alarm is cleared.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 6 Wait 5 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Expand the DataNode capacity.

Step 7 Expand the DataNode capacity.

Step 8 Wait 5 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **HDFS** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 20 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.97 ALM-14024 Tenant Space Usage Exceeds the Threshold

Description

The system checks the space usage (used space of each directory/space allocated to each directory) of each directory associated with a tenant every hour and compares the space usage of each directory with the threshold set for the directory. This alarm is generated when the space usage exceeds the threshold.

This alarm is cleared when the space usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14024	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
TenantName	Specifies the tenant for which the alarm is generated.
DirectoryName	Specifies the directory for which the alarm is generated.
Trigger condition	Specifies the threshold for triggering the alarm.

Impact on the System

This alarm is generated if the space usage of the tenant directory exceeds the custom threshold. File writing to the directory is not affected. If the used space exceeds the maximum storage space allocated to the directory, the HDFS fails to write data to the directory.

Possible Causes

- The alarm threshold is improperly configured.
- The space allocated to the tenant is improper.

Procedure

Check whether the alarm threshold is appropriate.

Step 1 View the alarm location information to obtain the tenant name and tenant directory for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose the **Tenant Resources** page, select the tenant for which the alarm is generated, and click **Resources**. Check whether the storage space threshold configured for the tenant directory for which the alarm is generated is proper. (The default value 90% is a proper value. You can set it based on the site requirements.)

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 On the **Resources** page, click **Modify** to modify or delete the storage space threshold.

Step 4 About one minute later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the space allocated to the tenant is appropriate.

Step 5 On the FusionInsight Manager portal, choose the **Tenant Resources** page, select the tenant for which the alarm is generated, and click **Resources**. Check whether the storage space quota of the tenant directory for which the alarm is generated is proper based on the actual service status of the tenant directory.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 On the **Resources** page, click **Modify** to modify the storage space quota.

Step 7 About one minute later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 9 Select **HDFS** in the required cluster and **NodeAgent** under **Manager** from the **Service**.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 20 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.98 ALM-14025 Tenant File Object Usage Exceeds the Threshold

Description

The system checks the file object usage (used file objects of each directory/ number of file objects allocated to each directory) of each directory associated with a tenant every hour and compares the file object usage of each directory with the threshold set for the directory. This alarm is generated when the file object usage exceeds the threshold.

This alarm is cleared when the file object usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14025	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
TenantName	Specifies the tenant for which the alarm is generated.
DirectoryName	Specifies the directory for which the alarm is generated.
Trigger condition	Specifies the threshold for triggering the alarm.

Impact on the System

This alarm is generated if the usage of file objects in a tenant directory exceeds the custom threshold. File writing to the directory is not affected. If the number of used file objects exceeds the maximum number of file objects allocated to the directory, the HDFS fails to write data to the directory.

Possible Causes

- The alarm threshold is improperly configured.
- The maximum number of file objects allocated to the tenant directory is inappropriate.

Procedure

Check whether the alarm threshold is appropriate.

Step 1 View the alarm location information to obtain the tenant name and tenant directory for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose the **Tenant Resources** page, select the tenant for which the alarm is generated, and click **Resources**. Check whether the file object threshold configured for the tenant directory for which the alarm is generated is proper. (The default value 90% is a proper value. You can set it based on the site requirements.)

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 On the **Resources** page, click **Modify** to modify or delete the file object threshold of the tenant directory for which the alarm is generated.

Step 4 About one minute later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the maximum number of file objects allocated to the tenant is appropriate.

Step 5 On the FusionInsight Manager portal, choose the **Tenant Resources** page, select the tenant for which the alarm is generated, and click **Resources**. Check whether the maximum number of file objects configured for the tenant directory for which the alarm is generated is proper based on the actual service status of the tenant directory.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 On the **Resources** page, click **Modify** to modify or delete the maximum number of file objects configured for the tenant directory.

Step 7 About one minute later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 9 Select **HDFS** in the required cluster and **NodeAgent** under **Manager** from the **Service**.

Step 10 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 20 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.99 ALM-14026 Blocks on DataNode Exceed the Threshold

Description

The system checks the number of blocks on each DataNode every 30 seconds. This alarm is generated when the number of blocks on the DataNode exceeds the threshold.

If the number of smoothing times is 1 and the number of blocks on the DataNode is less than or equal to the threshold, this alarm is cleared. If the number of smoothing times is greater than 1 and the number of blocks on the DataNode is less than or equal to 90% of the threshold, this alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
14026	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger condition	Specifies the threshold for triggering the alarm.

Impact on the System

If this alarm is reported, there are too many blocks on the DataNode. In this case, data writing into the HDFS may fail due to insufficient disk space.

Possible Causes

- The alarm threshold is improperly configured.
- Data skew occurs among DataNodes.
- The disk space configured for the HDFS cluster is insufficient.

Procedure

Modify the threshold

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services > HDFS** > **Configurations** > **All Configurations**. On the displayed page, find the **GC_OPTS** parameter under **HDFS->DataNode**.

Step 2 Set the threshold of the DataNode block number. Specifically, modify the value of **Xmx** of the **GC_OPTS** parameter. **Xmx** specifies the memory, and each GB memory supports a maximum of 500000 DataNode blocks. Set the memory as required. Confirm that **GC_PROFILE** is set to **custom** and save the configuration.

Step 3 Choose **Cluster >** *Name of the desired cluster* **> HDFS > Instance**, select the **DataNode** instance whose **Configuration Status** is **Expired**, and choose **More > Restart Instance** to make the **GC_OPTS** configuration take effect.

Step 4 Five minutes later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether associated alarms are reported.

Step 5 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms**, and check whether **ALM-14002 DataNode Disk Usage Exceeds the Threshold** is reported.

- If yes, go to **Step 6**.
- If no, go to **Step 8**.

Step 6 Rectify the fault by referring to **ALM-14002 DataNode Disk Usage Exceeds the Threshold**. Then, check whether the alarm is cleared.

- If yes, go to **Step 7**.
- If no, go to **Step 8**.

Step 7 Five minutes later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Expand the DataNode capacity.

Step 8 Expand the DataNode capacity.

Step 9 On the FusionInsight Manager portal, five minutes later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

Collect fault information.

Step 10 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 11 Select **HDFS** in the required cluster from the **Service**.

Step 12 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 20 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

Configuration rules of the DataNode JVM parameter

Default value of the DataNode JVM parameter **GC_OPTS**:

-Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=128M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -Djdk.tls.ephemeralDHKeySize=2048

The average number of blocks stored in each DataNode instance in the cluster is: HDFS Block x 3 ÷ Number of DataNodes. If the average number changes, you need to change **-Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M** in the default value. The following table lists the reference values.

Table 10-89 DataNode JVM configuration

Average Number of Blocks in a DataNode Instance	Reference Value
2,000,000	**-Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M**
5,000,000	**-Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G**

Xmx specifies the memory, and each GB memory supports a maximum of 500000 DataNode blocks. Set the memory as required.

10.13.100 ALM-14027 DataNode Disk Fault

Description

The system checks the disk status on DataNodes every 60 seconds. This alarm is generated when a disk is faulty.

After all faulty disks on the DataNode are recovered, you need to manually clear the alarm and restart the DataNode.

Attribute

Alarm ID	Alarm Severity	Auto Clear
14027	Major	No

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Failed Volumes	Specifies the list of faulty disks.

Impact on the System

If this alarm is reported, there are abnormal disk partitions on the DataNode. This may cause the loss of written files.

Possible Causes

- The hard disk is faulty.
- The disk permissions are configured improperly.

Procedure

Check whether a disk alarm is generated.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** and check whether **ALM-12014 Partition Lost** or **ALM-12033 Slow Disk Fault** exists.

- If yes, go to **Step 2**.
- If no, go to **Step 4**.

Step 2 Rectify the fault by referring to the handling procedure of **ALM-12014 Partition Lost** or **ALM-12033 Slow Disk Fault**. Then, check whether the alarm is cleared.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Wait 5 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Modify disk permissions.

Step 4 Choose **O&M** > **Alarm** > **Alarms** and view **Location** and **Additional Information** of the alarm to obtain the location of the faulty disk.

Step 5 Log in to the node for which the alarm is generated as user **root**. Go to the directory where the faulty disk is located, and run the **ll** command to check whether the permission of the faulty disk is **711** and whether the user is **omm**.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 Modify the permission of the faulty disk. For example, if the faulty disk is **data1**, run the following commands:

chown omm:wheel data1

chmod 711 data1

Step 7 In the alarm list on Manager, click **Clear** in the **Operation** column of the alarm to manually clear the alarm. Choose **Cluster** > **Services** > **HDFS** > **Instance**, select the DataNode, choose **More** > **Restart Instance**, wait for 5 minutes, and check whether a new alarm is reported.

- If no, no further action is required.
- If yes, go to **Step 8**.

Collect the fault information.

Step 8 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 9 Expand the **Service** drop-down list, and select **HDFS** and **OMS** for the target cluster.

Step 10 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 20 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system does not automatically clear this alarm and you need to manually clear the alarm.

Related Information

None

10.13.101 ALM-14028 Number of Blocks to Be Supplemented Exceeds the Threshold

Description

The system checks the number of blocks to be supplemented every 30 seconds and compares the number with the threshold. The number of blocks to be supplemented has a default threshold. This alarm is generated when the number of blocks to be supplemented exceeds the threshold.

You can change the threshold specified by **Blocks Under Replicated (NameNode)** by choosing **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **HDFS** > **File and Block**.

If **Trigger Count** is set to **1** and the number of blocks to be supplemented is less than or equal to the threshold, this alarm is cleared. If **Trigger Count** is greater than **1** and the number of blocks to be supplemented is less than or equal to 90% of the threshold, this alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
14028	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Data stored in HDFS is lost. HDFS may enter the security mode and cannot provide write services. Lost block data cannot be restored.

Possible Causes

- The DataNode instance is abnormal.
- Data is deleted.
- The number of replicas written into the file is greater than the number of DataNodes.

Procedure

Step 1 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, check whether alarm **ALM-14003 Number of Lost HDFS Blocks Exceeds the Threshold** is generated.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Rectify the fault according to the handling procedure of **ALM-14003 Number of Lost HDFS Blocks Exceeds the Threshold**. Five minutes later, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Step 3 Log in to the HDFS client as user **root**. The user password is defined by the user before the installation. Contact the system administrator to obtain the password. Run the following commands:

- Security mode:

  cd *Client installation directory*

  source bigdata_env

  kinit hdfs
- Normal mode:

  su - omm

  cd *Client installation directory*

  source bigdata_env

Step 4 Run the **hdfs fsck / >> fsck.log** command to obtain the status of the current cluster.

Step 5 Run the following command to count the number (*M*) of blocks to be replicated:

cat fsck.log | grep "Under-replicated"

Step 6 Run the following command to count the number (*N*) of blocks to be replicated in the **/tmp/hadoop-yarn/staging/** directory:

cat fsck.log | grep "Under replicated" | grep "/tmp/hadoop-yarn/staging/" | wc -l

NOTE

/tmp/hadoop-yarn/staging/ is the default directory. If the directory is modified, obtain it from the configuration item **yarn.app.mapreduce.am.staging-dir** in the **mapred-site.xml** file.

Step 7 Check whether the percentage of *N* is greater than 50% (N/M > 50%).

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Run the following command to reconfigure the number of file replicas in the directory (set the number of file replicas to the number of DataNodes or the default number of file replicas):

hdfs dfs -setrep -w *Number of file replicas*/tmp/hadoop-yarn/staging/

NOTE

To obtain the default number of file replicas:

Log in to FusionInsight Manager, choose **Cluster > Services > HDFS > Configurations > All Configurations**, and search for the **dfs.replication** parameter. The value of this parameter is the default number of file replicas.

Check whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Select **HDFS** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.102 ALM-14029 Number of Blocks in a Replica Exceeds the Threshold

Description

The system checks the number of blocks in a single replica every four hours and compares the number with the threshold. There is a threshold for the number of blocks in a single replica. This alarm is generated when the actual number of blocks in a single replica exceeds the threshold.

This alarm is cleared when the number of blocks to be supplemented is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
14029	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
NameServiceName	Specifies the NameService for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Replica data is prone to be lost when a node is faulty. Too many files of a single replica affect the security of the HDFS file system.

Possible Causes

- The DataNode is faulty.
- The disk is faulty.
- Files are written to a single replica.

Procedure

Step 1 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, check whether alarm **ALM-14003 Number of Lost HDFS Blocks Exceeds the Threshold** is generated.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Rectify the fault according to the handling procedure of **ALM-14003 Number of Lost HDFS Blocks Exceeds the Threshold**. In the next detection period, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Step 3 Check whether files of a single replica have been written into the service.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 Log in to the HDFS client as user **root**. The user password is defined by the user before the installation. Contact the system administrator to obtain the password. Run the following commands:

- Security mode:

  cd *Client installation directory*

  source bigdata_env

  kinit hdfs

- Normal mode:

  su - omm

  cd *Client installation directory*

  source bigdata_env

Step 5 Run the following command on the client node to increase the number of replicas for a single replica file:

hdfs dfs -setrep -w *file replica number file name or file path*

Step 6 In the next detection period, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect the fault information.

Step 7 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 8 Select **HDFS** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.103 ALM-16000 Percentage of Sessions Connected to the HiveServer to Maximum Number Allowed Exceeds the Threshold

Description

The system detects the percentage of sessions connected to the HiveServer to the maximum number of allowed sessions every 30 seconds. This indicator can be viewed on the **Cluster** > *Name of the desired cluster* > **Services** > **Hive > Instance** > *HiveServer instance***.** This alarm is generated when the percentage exceeds the default value **90%**.

To change the threshold, choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **Hive > Percentage of Sessions Connected to the HiveServer to Maximum Number of Sessions Allowed by the HiveServer**.

When the **Trigger Count** is 1, this alarm is cleared when the percentage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the percentage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16000	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If a connection alarm is generated, too many sessions are connected to Hive and new connections are unavailable.

Possible Causes

Too many clients are connected to HiveServer.

Procedure

Increase the maximum number of connections to Hive.

Step 1 On the FusionInsight Manager portal, Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations >All Configurations**.

Step 2 Search for **hive.server.session.control.maxconnections** and increase the value of this parameter. If the value of this parameter is **A**, the threshold is **B**, and the number of sessions connected to the HiveServer is **C**, adjust the value of this parameter according to **A x B > C**. To view the number of sessions connected to the HiveServer, check the value of **Statistics for Sessions of the HiveServer** on the Hive monitoring page.

Step 3 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Collect fault information.

Step 4 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 5 Select **Hive** in the required cluster from the **Service**.

Step 6 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.104 ALM-16001 Hive Warehouse Space Usage Exceeds the Threshold

Description

This alarm is generated when the Hive warehouse space usage exceeds the specified threshold (85% by default). The system checks the Hive data warehouse space usage every 30s. The indicator **Percentage of HDFS Space Used by Hive to the Available Space** can be viewed on the Hive service monitoring page.

To change the threshold, choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* **> Hive > Percentage of HDFS Space Used by Hive to the Available Space**.

When the **Trigger Count** is 1, this alarm is cleared when the Hive warehouse space usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the Hive warehouse space usage is less than or equal to 90% of the threshold.

NOTE

The administrator can reduce the warehouse space usage by expanding the warehouse capacity or releasing the used space.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16001	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The system fails to write data, which causes data loss.

Possible Causes

- The upper limit of the HDFS capacity available for Hive is too small.
- The HDFS space is insufficient.
- Some data nodes break down.

Procedure

Expand the system configuration.

Step 1 Analyze the cluster HDFS capacity usage and increase the upper limit of the HDFS capacity available for Hive.

Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations > All Configurations**, find **hive.metastore.warehouse.size.percent**, and increase its value so that larger HDFS capacity will be available for Hive. Assume that the value of the configuration item is A, the total HDFS storage space is B, the threshold is C, and the HDFS space used by Hive is D. The adjustment policy is A x B x C > D. The total HDFS storage space can be viewed on the HDFS NameNode page. The HDFS space used by Hive can be viewed on the Hive monitoring page.

Step 2 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Expand the system.

Step 3 Expand the system.

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the data node is normal.

Step 5 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms**.

Step 6 Check whether "ALM-12006 Node Fault", "ALM-12007 Process Fault", or "ALM-14002 DataNode Disk Usage Exceeds the Threshold" exist.

- If yes, go to **Step 7**.
- If no, go to **Step 9**.

Step 7 Clear the alarm by following the steps provided in "ALM-12006 Node Fault", "ALM-12007 Process Fault", and "ALM-14002 DataNode Disk Usage Exceeds the Threshold".

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **Hive** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.105 ALM-16002 Hive SQL Execution Success Rate Is Lower Than the Threshold

Description

The system checks the percentage of the HQL statements that are executed successfully in every 30 seconds. The formula is: Percentage of HQL statements that are executed successfully = Number of HQL statements that are executed successfully by Hive in a specified period/Total number of HQL statements that

are executed by Hive. This indicator can be viewed on the **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance** > *HiveServer instance* **.** The default threshold of the percentage of HQL statements that are executed successfully is **90%**. An alarm is reported when the percentage is lower than the **90%**. Users can view the name of the host where an alarm is generated in the location information about the alarm. The IP address of the host is the IP address of the HiveServer node.

Users can modify the threshold by choosing **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Hive** > **Percentage of HQL Statements That Are Executed Successfully by Hive**.

This alarm is cleared when the execution success rate is higher than 110% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16002	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The system configuration and performance cannot meet service processing requirements.

Possible Causes

- A syntax error occurs in HQL statements.
- The HBase service is abnormal when a Hive on HBase task is performed.

- The Spark service is abnormal when a Hive on Spark task is performed.
- The dependent basic services, such as HDFS, Yarn, and ZooKeeper, are abnormal.

Procedure

Check whether the HQL statements comply with syntax.

Step 1 On the FusionInsight Manager page, choose **O&M** > **Alarm** to view the alarm details and obtain the node where the alarm is generated.

Step 2 Use the Hive client to log in to the HiveServer node where an alarm is reported. Query the HQL syntax provided by Apache, and check whether the HQL commands are correct. For details, see **https://cwiki.apache.org/confluence/display/hive/languagemanual**.

- If yes, go to **Step 4**.
- If no, go to **Step 3**.

NOTE

To view the user who runs an incorrect statement, you can download the hiveserver audit log file of the HiveServer node where this alarm is generated. **Start Data** and **End Data** are 10 minutes before and after the alarm generation time respectively. Open the log file and search for the **Result=FAIL** keyword to filter the log information about the incorrect statement, and then view the user who runs the incorrect statement according to **UserName** in the log information.

Step 3 Enter the correct HQL statements, and check whether the command can be properly executed.

- If yes, go to **Step 12**.
- If no, go to **Step 4**.

Check whether the HBase service is abnormal.

Step 4 Check whether an Hive on HBase task is performed with the user who runs the HQL command.

- If yes, go to **Step 5**.
- If no, go to **Step 8**.

Step 5 On the FusionInsight Manager page, click **Cluster** > *Name of the desired cluster* > **Services**, check whether the HBase service is normal in the service list.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 Choose **O&M** > **Alarm**, check the related alarms displayed on the alarm page and clear them according to related alarm help.

Step 7 Enter the correct HQL statements, and check whether the command can be properly executed.

- If yes, go to **Step 12**.
- If no, go to **Step 8**.

Check whether the HDFS, Yarn, and ZooKeeper are normal.

Step 8 On the FusionInsight Manager portal, click **Cluster** > *Name of the desired cluster* > **Services**.

Step 9 In the service list, check whether the services, such as HDFS, Yarn, and ZooKeeper are normal.

- If yes, go to **Step 12**.
- If no, go to **Step 10**.

Step 10 Check the related alarms displayed on the alarm page and clear them according to related alarm help.

Step 11 Enter the correct HQL statements, and check whether the command can be properly executed.

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 After 1 minute, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collect fault information.

Step 13 On the FusionInsight Manager home page, choose **O&M** > **Log** > **Download**.

Step 14 Select the following nodes in the required cluster from the **Service**:

- MapReduce
- Hive

Step 15 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.106 ALM-16003 Background Thread Usage Exceeds the Threshold

Description

The system checks the background thread usage in every 30 seconds. This alarm is generated when the usage of the background thread pool of Hive exceeds the threshold, 90% by default.

NOTE

MRS 3.X supports the multi-instance function. If the multi-instance function is enabled in the cluster and multiple Hive services are installed, determine the Hive service for which the alarm is generated based on the value of **ServiceName** in **Location** of the alarm. For example, if Hive1 service is unavailable, **ServiceName** is set to **Hive1** in **Location**, and the operation object in the handling procedure is changed from Hive to Hive1.

Attribute

Alarm ID	Alarm Severity	Auto Clear
16003	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

There are too many background threads, so the newly submitted task cannot run in time.

Possible Causes

The usage of the background thread pool of Hive is excessively high when:

- There are many tasks executed in the background thread pool of HiveServer.
- The capacity of the background thread pool of HiveServer is too small.

Procedure

Check the number of tasks executed in the background thread pool of HiveServer.

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive**. On the displayed page, click **HiveServer Instance** and check values of **Background Thread Count** and **Background Thread Usage**.

Step 2 Check whether the number of background threads in the latest half an hour is excessively high. (By default, the queue number is 100, and the thread number is considered as high if it is 90 or larger.)

- If it is, go to **Step 3**.
- If it is not, go to **Step 5**.

Step 3 Adjust the number of tasks submitted to the background thread pool. (For example, cancel some time-consuming tasks with low performance.)

Step 4 Check whether the values of Background Thread Count and Background Thread Usage decrease.

- If it is, go to **Step 7**.
- If it is not, go to **Step 5**.

Check the capacity of the HiveServer background thread pool.

Step 5 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive**. On the displayed page, click **HiveServer Instance** and check values of Background Thread Count and Background Thread Usage.

Step 6 Increase the value of **hive.server2.async.exec.threads** in the **${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/1_23_HiveServer/etc/hive-site.xml** file. For example, increase the value by 20%.

Step 7 Save the modification.

Step 8 Check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **Hive** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.107 ALM-16004 Hive Service Unavailable

Description

This alarm is generated when the HiveServer service is unavailable. The system checks the HiveServer service status every 60 seconds.

This alarm is cleared when the HiveServer service is normal.

NOTE

MRS 3.X supports the multi-instance function. If the multi-instance function is enabled in the cluster and multiple Hive service instances are installed, you need to determine the Hive service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the Hive1 service is unavailable, **ServiceName=Hive1** is displayed in **Location**, and the operation object in the procedure needs to be changed from Hive to Hive1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16004	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The system cannot provide data loading, query, and extraction services.

Possible Causes

- Hive service unavailability may be related to the faults of the Hive process as well as basic services, such as ZooKeeper, Hadoop distributed file system (HDFS), Yarn, and DBService.
  - The ZooKeeper service is abnormal.
  - The HDFS service is abnormal.

  - The Yarn service is abnormal.
  - The DBService service is abnormal.
  - The Hive service process is abnormal. If the alarm is caused by Hive process fault, the alarm report has a delay of about 5 minutes.
- The network communication between the Hive and basic services is interrupted.

Procedure

Check the HiveServer/MetaStore process status.

Step 1 On the FusionInsight Manager portal, click **Cluster >** *Name of the desired cluster* **> Services** > **Hive** > **Instance**. In the Hive instance list, check whether the HiveServer or MetaStore instances are in the Unknown state.

- If yes, go to **Step 2**.
- If no, go to **Step 4**.

Step 2 In the Hive instance list, choose **More** > **Restart Instance** to restart the HiveServer/MetaStore process.

Step 3 In the alarm list, check whether **Hive Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check the ZooKeeper service status.

Step 4 On the FusionInsight Manager, check whether the alarm list contains **Process Fault**.

- If yes, go to **Step 5**.
- If no, go to **Step 8**.

Step 5 In the **Process Fault**, check whether **ServiceName** is **ZooKeeper**.

- If yes, go to **Step 6**.
- If no, go to **Step 8**.

Step 6 Rectify the fault by following the steps provided in "ALM-12007 Process Fault".

Step 7 In the alarm list, check whether **Hive Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check the HDFS service status.

Step 8 On the FusionInsight Manager, check whether the alarm list contains **HDFS Service Unavailable**.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

Step 9 Rectify the fault by following the steps provided in "ALM-14000 HDFS Service Unavailable".

Step 10 In the alarm list, check whether **Hive Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Check the Yarn service status.

Step 11 In FusionInsight Manager alarm list, check whether **Yarn Service Unavailable** is generated.

- If yes, go to **Step 12**.
- If no, go to **Step 14**.

Step 12 Rectify the fault. For details, see "ALM-18000 Yarn Service Unavailable".

Step 13 In the alarm list, check whether **Hive Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Check the DBService service status.

Step 14 In FusionInsight Manager alarm list, check whether **DBService Service Unavailable** is generated.

- If yes, go to **Step 15**.
- If no, go to **Step 17**.

Step 15 Rectify the fault. For details, see "ALM-27001 DBService Service Unavailable".

Step 16 In the alarm list, check whether **Hive Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 17**.

Check the network connection between the Hive and ZooKeeper, HDFS, Yarn, and DBService.

Step 17 On the FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Services** > **Hive**.

Step 18 Click **Instance**.

The HiveServer instance list is displayed.

Step 19 Click **Host Name** in the row of **HiveServer**.

The active HiveServer host status page is displayed.

Step 20 Record the IP address under **Basic Information**.

Step 21 Use the IP address obtained in **Step 20** to log in to the host where the active HiveServer runs as user **omm**.

Step 22 Run the **ping** command to check whether communication between the host that runs the active HiveServer and the hosts that run the ZooKeeper, HDFS, Yarn, and DBService services is normal. (Obtain the IP addresses of the hosts that run the ZooKeeper, HDFS, Yarn, and DBService services in the same way as that for obtaining the IP address of the active HiveServer.)

- If yes, go to **Step 25**.
- If no, go to **Step 23**.

Step 23 Contact the administrator to restore the network.

Step 24 In the alarm list, check whether **Hive Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 25**.

Collect fault information.

Step 25 On the FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 26 Select the following nodes in the required cluster from the **Service**:

- ZooKeeper
- HDFS
- Yarn
- DBService
- Hive

Step 27 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 28 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.108 ALM-16005 The Heap Memory Usage of the Hive Process Exceeds the Threshold

Description

The system checks the Hive service status every 30 seconds. The alarm is generated when the heap memory usage of an Hive service exceeds the threshold (95% of the maximum memory).

Users can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Hive** to change the threshold.

The alarm is cleared when the heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

When the heap memory usage of Hive is overhigh, the performance of Hive task operation is affected. In addition, a memory overflow may occur so that the Hive service is unavailable.

Possible Causes

The heap memory of the Hive instance on the node is overused or the heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms** and select the alarm whose **Alarm ID** is **16005**. Then check the role name in **Location** and confirm the IP adress of the instance.

- If the role for which the alarm is generated is HiveServer, go to **Step 2**.
- If the role for which the alarm is generated is MetaStore, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* **> Services** > **Hive** > **Instance** and click the HiveServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory**, and select **HiveServer Memory Usage Statistics** and click **OK**, check whether the used heap memory of the HiveServer service reaches the threshold(default value: 95%) of the maximum heap memory specified for HiveServer.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance** and click the MetaStore for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory**, and select **MetaStore Memory Usage Statistics** and click **OK**, check whether the used heap memory of the MetaStore service reaches the threshold(default value: 95%) of the maximum heap memory specified for MetaStore.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations > All Configurations**. Choose **HiveServer/MetaStore** > **JVM**. Adjust the value of **-Xmx** in **HIVE_GC_OPTS/ METASTORE_GC_OPTS** as the following rules. Click **Save**.

NOTE

Suggestions for GC parameter settings for the HiveServer:

- When the heap memory used by the HiveServer process reaches the threshold (default value: 95%) of the maximum heap memory set by the HiveServer process, change the value of **-Xmx** to twice the default value. For example, if **-Xmx** is set to 2GB by default, change the value of **-Xmx** to 4GB. You are advised to change the value of **-Xms** to set the ratio of **-Xms** and **-Xmx** to 1:2 to avoid performance problems when JVM dynamically. On the FusionInsight Manager home page, choose **O&M**> **Alarm**> **Thresholds** > *Name of the desired cluster* **> Hive** > **CPU and Memory** > **HiveServer Heap Memory Usage Statistics (HiveServer)** to view **Threshold**.

Suggestions for GC parameter settings for the MetaServer:

- When the heap memory used by the MetaStore process reaches the threshold (default value: 95%) of the maximum heap memory set by the MetaStore process, change the value of **-Xmx** to twice the default value. For example, if **-Xmx** is set to 2GB by default, change the value of **-Xmx** to 4GB. On the FusionInsight Manager home page, choose **O&M**> **Alarm**> **Thresholds** > *Name of the desired cluster* **> Hive** > **CPU and Memory** > **MetaStore Heap Memory Usage Statistics (MetaStore)** to view **Threshold**.
- You are advised to change the value of **-Xms** to set the ratio of **-Xms** and **-Xmx** to 1:2 to avoid performance problems when JVM dynamically.

Step 5 Click **More > Restart Service** to restart the service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Hive** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.109 ALM-16006 The Direct Memory Usage of the Hive Process Exceeds the Threshold

Description

The system checks the Hive service status every 30 seconds. The alarm is generated when the direct memory usage of an Hive service exceeds the threshold (95% of the maximum memory).

Users can choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* **> Hive** to change the threshold.

The alarm is cleared when the direct memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the direct memory usage of Hive is overhigh, the performance of Hive task operation is affected. In addition, a memory overflow may occur so that the Hive service is unavailable.

Possible Causes

The direct memory of the Hive instance on the node is overused or the direct memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check direct memory usage.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms** and select the alarm whose **Alarm ID** is **16006**. Then check the role name in **Location** and confirm the IP adress of the instance.

- If the role for which the alarm is generated is HiveServer, go to **Step 2**.
- If the role for which the alarm is generated is MetaStore, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services > Hive > Instance** and click the HiveServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory**, and select **HiveServer Memory Usage Statistics** and click **OK**, check whether the used direct memory of the HiveServer service reaches the threshold(default value: 95%) of the maximum direct memory specified for HiveServer.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance** and click the MetaStore for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory**, and select **MetaStore Memory Usage Statistics** and click **OK**, check whether the used direct memory of the MetaStore service reaches the threshold(default value: 95%) of the maximum direct memory specified for MetaStore.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** >*Name of the desired cluster* > **Services** > **Hive** > **Configurations > All Configurations**. Choose

HiveServer/MetaStore > **JVM**. Adjust the value of **-XX:MaxDirectMemorySize** in **HIVE_GC_OPTS/METASTORE_GC_OPTS** as the following rules. Click **Save**.

NOTE

Suggestions for GC parameter settings for the HiveServer:

- It is recommended that you set the value of **-XX:MaxDirectMemorySize** to 1/8 of the value of **-Xmx**. For example, if **-Xmx** is set to 8 GB, **-XX:MaxDirectMemorySize** is set to 1024 MB. If **-Xmx** is set to 4 GB, **-XX:MaxDirectMemorySize** is set to 512 MB. It is recommended that the value of **-XX:MaxDirectMemorySize** be greater than or equal to 512 MB.

Suggestions for GC parameter settings for the MetaServer:

- It is recommended that you set the value of **-XX:MaxDirectMemorySize** to 1/8 of the value of **-Xmx**. For example, if **-Xmx** is set to 8 GB, **-XX:MaxDirectMemorySize** is set to 1024 MB. If **-Xmx** is set to 4 GB, **-XX:MaxDirectMemorySize** is set to 512 MB. It is recommended that the value of **-XX:MaxDirectMemorySize** be greater than or equal to 512 MB.

Step 5 Click **More > Restart Service** to restart the service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Hive** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.110 ALM-16007 Hive GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) time of the Hive service every 60 seconds. This alarm is generated when the detected GC time exceeds the threshold (exceeds 12 seconds for three consecutive checks.) To change the threshold, choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **Hive**. This alarm is cleared when the Hive GC time is shorter than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the GC time exceeds the threshold, Hive data read and write are affected.

Possible Causes

The memory of Hive instances is overused, the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC time.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms** and select the alarm whose **Alarm ID** is **16007**. Then check the role name in **Location** and confirm the IP adress of the instance.

- If the role for which the alarm is generated is HiveServer, go to **Step 2**.
- If the role for which the alarm is generated is MetaStore, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** >*Name of the desired cluster*> **Services** > **Hive** > **Instance** and click the HiveServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **GC**, and select **Garbage Collection (GC)**

Time of HiveServer and click **OK** to check whether the GC time is longer than 12 seconds.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** >*Name of the desired cluster* > **Services** > **Hive** > **Instance** and click the MetaStore for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **GC**, and select **Garbage Collection (GC) Time of MetaStore** and click **OK** to check whether the GC time is longer than 12 seconds.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Check the current JVM configuration.

Step 4 On the FusionInsight Manager portal, choose **Cluster** >*Name of the desired cluster* > **Services > Hive > Configurations > All Configurations**. Choose **HiveServer/MetaStore** > **JVM**. Adjust the value of **-Xmx** in **HIVE_GC_OPTS/ METASTORE_GC_OPTS** as the following rules. Click **Save**.

NOTE

Suggestions for GC parameter settings for the HiveServer:

- When the Hive GC time exceeds the threshold, change the value of **-Xmx** to twice the default value. For example, if **-Xmx** is set to 2 GB by default, change the value of **-Xmx** to 4 GB.
- You are advised to change the value of **-Xms** to set the ratio of **-Xms** and **-Xmx** to 1:2 to avoid performance problems when JVM dynamically.

Suggestions for GC parameter settings for the MetaServer:

- When the Meta GC time exceeds the threshold, change the value of **-Xmx** to twice the default value. For example, if **-Xmx** is set to 2 GB by default, change the value of **-Xmx** to 4 GB.
- You are advised to change the value of **-Xms** to set the ratio of **-Xms** and **-Xmx** to 1:2 to avoid performance problems when JVM dynamically.

Step 5 Click **More > Restart Service** to restart the service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal of active and standby clusters, choose **O&M** > **Log > Download**.

Step 8 In the **Service**, select **Hive** in the required cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.111 ALM-16008 Non-Heap Memory Usage of the Hive Process Exceeds the Threshold

Description

The system checks the Hive service status every 30 seconds. The alarm is generated when the non-heap memory usage of an Hive service exceeds the threshold (95% of the maximum memory).

Users can choose **O&M > Alarm > Thresholds >** *Name of the desired cluster* > **Hive** to change the threshold.

The alarm is cleared when the non-heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
16008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

When the non-heap memory usage of Hive is overhigh, the performance of Hive task operation is affected. In addition, a memory overflow may occur so that the Hive service is unavailable.

Possible Causes

The non-heap memory of the Hive instance on the node is overused or the non-heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check non-heap memory usage.

Step 1 On the FusionInsight Manager portal, click **O&M > Alarm > Alarms** and select the alarm whose **Alarm ID** is **16008**. Then check the role name in **Location** and confirm the IP adress of the instance.

- If the role for which the alarm is generated is HiveServer, go to **Step 2**.
- If the role for which the alarm is generated is MetaStore, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance** and click the HiveServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory**, and select **HiveServer Memory Usage Statistics** and click **OK**, check whether the used non-heap memory of the HiveServer service reaches the threshold(default value: 95%) of the maximum non-heap memory specified for HiveServer.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance** and click the MetaStore for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory**, and select **MetaStore Memory Usage Statistics** and click **OK**, check whether the used non-heap memory of the MetaStore service reaches the threshold(default value: 95%) of the maximum non-heap memory specified for MetaStore.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Hive** > **Configurations > All Configurations**. Choose **HiveServer/MetaStore** > **JVM**. Adjust the value of **-XX:MaxMetaspaceSize** in **HIVE_GC_OPTS/METASTORE_GC_OPTS** as the following rules. Click **Save**.

NOTE

Suggestions for GC parameter settings for the HiveServer:

- It is recommended that you set the value of **-XX:MaxMetaspaceSize** to 1/8 of the value of **-Xmx**. For example, if **-Xmx** is set to 2 GB, **-XX:**

  MaxMetaspaceSize is set to 256 MB. If **-Xmx** is set to 4 GB, **-XX:MaxMetaspaceSize** is set to 512 MB.

Suggestions for GC parameter settings for the MetaServer:

- It is recommended that you set the value of **-XX:MaxMetaspaceSize** to 1/8 of the value of **-Xmx**. For example, if **-Xmx** is set to 2 GB, **-XX:**

  MaxMetaspaceSize is set to 256 MB. If **-Xmx** is set to 4 GB, **-XX:MaxMetaspaceSize** is set to 512 MB

Step 5 Click **More > Restart Service** to restart the service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Hive** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.112 ALM-16009 Map Number Exceeds the Threshold

Description

The system checks the number of HQL maps in every 30 seconds. This alarm is generated if the number exceeds the threshold. By default, **Trigger Count** is set to **3**, and the threshold is 5000.

Attribute

Alarm ID	Alarm Severity	Auto Clear
16009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the number of HQL maps executed on Hive is excessively large, the HQL execution speed is slow, and a large number of resources are occupied.

Possible Causes

The HQL statements are not the optimal.

Procedure

Check the number of HQL maps.

Step 1 On FusionInsight Manager portal, choose **Cluster** >*Name of the desired cluster*> **Services** > **Hive** > **Resource**. Check the HQL statements with the excessively large number (5000 or more) of maps in **HQL Map Count**.

Step 2 Locate the corresponding HQL statements, optimize them and execute them again.

Step 3 Check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 On the FusionInsight Manager, choose **O&M** > **Log > Download**.

Step 5 Select **Hive** in the required cluster from the **Service**.

Step 6 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.113 ALM-16045 Hive Data Warehouse Is Deleted

Description

The system checks the Hive data warehouse in every 60 seconds.This alarm is generated when the Hive data warehouse is deleted.

Attribute

Alarm ID	Alarm Severity	Auto Clear
16045	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The default Hive data warehouse is deleted. As a result, creating databases or tables in the default data warehouse fails, and services are affected.

Possible Causes

Hive periodically checks the status of the default data warehouse and finds that the default data warehouse is deleted.

Procedure

Check the default Hive data warehouse.

Step 1 Log in to the node where the client is located as user **root**.

Step 2 Run the following command to check whether the **warehouse** directory exists in **hdfs://hacluster/user/***<username>***/.Trash/Current/**.

hdfs dfs -ls hdfs://hacluster/user/*<username>*/.Trash/Current/

For example, if **user/hive/warehouse** exists:

```
host01:/opt/Bigdata/client # hdfs dfs -ls hdfs://hacluster/user/test/.Trash/Current/
Found 1 items
drwx------   - test hadoop          0 2019-06-17 19:53 hdfs://hacluster/user/test/.Trash/Current/user
```

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 By default, there is an automatic recovery mechanism for the data warehouse. You can wait for 5 ~10s to check whether the default data warehouse is restored. If the data warehouse is not recovered, manually run the following command to restore the data warehouse.

hdfs dfs -mv hdfs://hacluster/user/*<username>*/.Trash/Current/user/hive/warehouse /user/hive/warehouse

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect fault information.

Step 5 Collect related information in the **.Trash/Current/** directory on the client background.

Step 6 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.114 ALM-16046 Hive Data Warehouse Permission Is Modified

Description

The system checks the Hive data warehouse permission in every 60 seconds. This alarm is generated if the permission is modified.

Attribute

Alarm ID	Alarm Severity	Auto Clear
16046	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

If the permission on the Hive default data warehouse is modified, the permission for users or user groups to create databases or tables in the default data warehouse is changed.

Possible Causes

Hive periodically checks the status of the default data warehouse and finds that default data warehouse permission is changed.

Procedure

Check the Hive default data warehouse permission.

Step 1 Log in to the node where the client is located as user **root**.

Step 2 Restore the directory permission based on the current cluster as a user with the **supergroup** permission.

- In security mode, run the following command: **hdfs dfs -chmod 770 hdfs://hacluster/user/hive/warehouse**
- In non-security mode, run the following command: **hdfs dfs -chmod 777 hdfs://hacluster/user/hive/warehouse**

Step 3 Check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 4**.

Collect fault information.

Step 4 Collect related information in the **hdfs://hacluster/user/hive/warehouse** directory on the client background.

Step 5 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.115 ALM-16047 HiveServer Has Been Deregistered from ZooKeeper

Description

The system checks the Hive service every 60 seconds. This alarm is generated when Hive registration information on ZooKeeper is lost or Hive cannot connect to ZooKeeper.

Attribute

Alarm ID	Alarm Severity	Auto Clear
16047	Major	Yes

Parameters

Parameter	Meaning
Cluster Name	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

If the Hive configuration cannot be read from ZooKeeper, HiveServer will be unavailable.

Possible Causes

- The network is disconnected.
- The ZooKeeper instance is abnormal.

Procedure

Restart related instances.

Step 1 Log in to FusionInsight Manager. On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Alarms**, click the drop-down list in the row that contains the alarm. Then check the role name in **Location** and confirm the IP adress of the instance.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance**, select the instance corresponding to the IP address for which the alarm is generated, and choose **More** > **Restart Instance**.

Step 3 Wait for 5 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If not, go to **Step 4**.

Collect the fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Expand the **Service** drop-down list, and select **Hive** for the target cluster.

Step 6 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearance

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.116 ALM-16048 Tez or Spark Library Path Does Not Exist

Description

The system checks the Tez and Spark library paths every 180 seconds. This alarm is generated when the Tez or Spark library path does not exist.

Attribute

Alarm ID	Alarm Severity	Auto Clear
16048	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Hive on Tez and Hive on Spark functions are affected.

Possible Causes

The Tez or Spark library path is deleted from the HDFS.

Procedure

Check the default Hive data warehouse.

Step 1 Log in to the node where the client is located as user **root**.

Step 2 Run the following command to check whether the **tezlib** or **sparklib** directory exists in the **hdfs://hacluster/user/{User name}/.Trash/Current/** director:

hdfs dfs -ls hdfs://hacluster/user/*<username>***/.Trash/Current/**

For example, the following information shows that **/user/hive/tezlib/8.1.0.1/** and **/user/hive/sparklib/8.1.0.1/** exist.

```
host01:/opt/Bigdata/client # hdfs dfs -ls hdfs://hacluster/user/test/.Trash/Current/
Found 1 items
drwx------   - test hadoop          0 2019-06-17 19:53 hdfs://hacluster/user/test/.Trash/Current/user
```

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Run the following command to restore **tezlib** and **sparklib**.

hdfs dfs -mv hdfs://hacluster/user/*<username>***/.Trash/Current/user/hive/tezlib/8.1.0.1/tez.tar.gz /user/hive/tezlib/8.1.0.1/tez.tar.gz**

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect fault information.

Step 5 Collect related information in the **.Trash/Current/** directory on the client background.

Step 6 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.117 ALM-17003 Oozie Service Unavailable

Description

The system checks the Oozie service status in every 5 seconds. This alarm is generated when Oozie or a component on which Oozie depends cannot provide services properly.

This alarm is automatically cleared when the Oozie service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
17003	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Oozie cannot be used to submit jobs.

Possible Causes

- The DBService service is abnormal or the data of Oozie stored in DBService is damaged.
- The HDFS service is abnormal or the data of Oozie stored in HDFS is damaged.
- The Yarn service is abnormal.
- The Nodeagent process is abnormal.

Procedure

Query the Oozie service health status code.

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* >**Services** > **Oozie**. Click **oozie** (any one is OK) on the **oozie WebUI**. to go to the Oozie WebUI.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

Step 2 Add **/servicehealth** to the URL in the address box of the browser and access again. The value of **statusCode** is the current Oozie service health status code.

For example, visit **https://10.10.0.117:20026/Oozie/oozie/130/oozie/servicehealth**. The result is as follows:

```
{"beans":[{"name":"serviceStatus","statusCode":0}]}
```

If the health status code cannot be displayed or the browser does not respond, the service may be unavailable due to Oozie process fault. See **Step 13** to rectify the fault.

Step 3 Perform the operations based on the error code. For details, see **Table 10-90**.

Table 10-90 Oozie service health status code

Status Code	Description	Error Cause	Solution
0	The service is running properly.	None	None
18002	The DBService service is abnormal.	Oozie fails to connect to DBService or the data stored in DBService is damaged.	See **Step 4**.

Status Code	Description	Error Cause	Solution
18003	The HDFS service is abnormal.	Oozie fails to connect to HDFS or the data stored in HDFS is damaged.	See **Step 7**.
18005	The MapReduce service is abnormal.	The Yarn service is abnormal.	See **Step 11**.

Check the DBService service.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services**, and check whether the DBService service is running properly.

- If yes, go to **Step 6**.
- If no, go to **Step 5**.

Step 5 Resolve the problem of DBService based on the alarm help and check whether the Oozie alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Step 6 Log in to the Oozie database to check whether the data is complete.

1. Log in to the active DBService node as user **root**.

   On the FusionInsight Manager page, choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService > Instance** to view the IP address of the active DBservice node.

2. Run the following command to log in to the Oozie database:

   su - omm

   source ${BIGDATA_HOME}/FusionInsight_BASE_8.1.0.1/install/FusionInsight-dbservice-2.7.0/.dbservice_profile

   gsql -U *Username* **-W** *Oozie database password* **-p 20051 -d** *Database name*

3. After the login is successful, enter **\d** to check whether there are 15 data tables.

   The Oozie service has 15 data tables by default. If these data tables are deleted or the table structure is modified, the Oozie service may be unavailable. Contact the O&M personnel to back up the data and perform restoration.

Check the HDFS service.

Step 7 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services**, and check whether the HDFS service is running properly.

- If yes, go to **Step 9**.
- If no, go to **Step 8**.

Step 8 Resolve the problem of HDFS based on the alarm help and check whether the Oozie alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Step 9 Log in to HDFS to check whether the Oozie file directory structure is complete.

1. Download and install an HDFS client..
2. Log in to the client node as user **root** and run the following commands to check whether **/user/oozie/share** exists.

   If the cluster uses the security mode, perform security authentication.

   kinit admin

   hdfs dfs -ls /user/oozie/share

- If yes, go to **Step 18**.
- If no, go to **Step 10**.

Step 10 In the Oozie client installation directory, manually upload the share directory to **/user/oozie** in HDFS, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Check the Yarn and MapReduce service.

Step 11 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services**, and check whether the Yarn and MapReduce services are running properly.

- If yes, go to **Step 18**.
- If no, go to **Step 12**.

Step 12 Resolve the problem of Yarn and MapReduce based on the alarm help and check whether the Oozie alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Check the Oozie process.

Step 13 Log in to each node of Oozie as user **root**.

Step 14 Run the **ps -ef | grep oozie** command to check whether the Oozie process exists.

- If yes, go to **Step 15**.
- If no, go to **Step 18**.

Step 15 Collect fault information in **prestartDetail.log**, **oozie.log**, and **catalina.out** in the Oozie log directory **/var/log/Bigdata/oozie**. If the alarm is not caused by manual misoperation, go to **Step 16**.

Check the Nodeagent process.

Step 16 Log in to each node of Oozie as user **root**. Run the **ps -ef | grep nodeagent** command to check whether the Nodeagent process exists.

- If yes, go to **Step 17**.
- If no, go to **Step 18**.

Step 17 Run the **kill -9** *The process ID of nodeagent* command, wait 10 minutes, and check whether alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Step 18 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.118 ALM-17004 Oozie Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of the Oozie service every 60 seconds. The alarm is generated when the heap memory usage of a Metadata instance exceeds the threshold (95% of the maximum memory). The alarm is cleared when the heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
17004	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The heap memory overflow may cause a service breakdown.

Possible Causes

The heap memory of the Oozie instance is overused or the heap memory is inappropriately allocated.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **Oozie Heap Memory Usage Exceeds the Threshold** > **Location**. Check the IP address of the instance involved in this alarm.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Oozie** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down menu in the chart area and choose **Customize** > **Memory** > **Oozie Heap Memory Resource Percentage**. Click **OK**.

Step 3 Check whether the used heap memory of Oozie reaches the threshold (the default value is 95% of the maximum heap memory) specified for Oozie.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Oozie** > **Configurations > All Configurations**. Set Search **GC_OPTS** in the search box. Increase the value of **-Xmx** as required, and click **Save** > **OK**.

NOTE

Suggestions on GC parameter settings for Oozie:

You are advised to set **-Xms** and **-Xmx** to the same value to prevent adverse impact on performance when JVM dynamically adjusts the heap memory size.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **Oozie** in the required cluster from the **Service**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.119 ALM-17005 Oozie Non Heap Memory Usage Exceeds the Threshold

Description

The system checks the non heap memory usage of Oozie every 30 seconds. This alarm is reported if the non heap memory usage of Oozie exceeds the threshold (80%). This alarm is cleared if the non heap memory usage is lower than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
17005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Non-heap memory overflow may cause service breakdown.

Possible Causes

The non-heap memory of the Oozie instance is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **Oozie Non Heap Memory Usage Exceeds the Threshold**. On the displayed page, check the location information of the alarm. Check the name of the instance host for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the target cluster* > **Services** > **Oozie** and click the **Instance** tab. On the displayed page, select the role corresponding to the host name for which the alarm is generated and select **Customize** from the drop-down list in the upper right corner of the chart area. Choose **Memory** and select **Oozie Non Heap Memory Resource Percentage**. Click **OK**.

Step 3 Check whether the non-heap memory used by Oozie reaches the threshold (80% of the maximum non-heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the target cluster* > **Services** > **Oozie** and click the **Configurations** and then **All Configurations**. On the displayed page, search for the **GC_OPTS** parameter in the search box and check whether it contains **-XX: MaxMetaspaceSize**. If yes, increase the value of **-XX: MaxMetaspaceSize** based on the site requirements. If no, manually add **-XX: MaxMetaspaceSize** and set its value to 1/8 of the value of **-Xmx**. Click **Save**, and then click **OK**

NOTE

JDK1.8 does not support the **MaxPermSize** parameter.

Suggestions on GC parameter settings for Oozie:

Set the value of **-XX:MaxMetaspaceSize** to 1/8 of the value of **-Xmx**. For example, if **-Xmx** is set to 2 GB, **-XX:MaxMetaspaceSize** is set to 256 MB. If **-Xmx** is set to 4 GB, **-XX:MaxMetaspaceSize** is set to 512 MB.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Oozie** for the target cluster.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.120 ALM-17006 Oozie Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the Oozie service every 30 seconds. The alarm is generated when the direct memory usage of an Oozie instance exceeds the threshold (80% of the maximum memory). The alarm is cleared when the direct memory usage of Oozie is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
17006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The direct memory overflow may cause a service breakdown.

Possible Causes

The direct memory of the Oozie instance is overused or the direct memory is inappropriately allocated.

Procedure

Check direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **Oozie Direct Memory Usage Exceeds the Threshold** > **Location**. Check the IP address of the instance involved in this alarm.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Oozie** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down menu in the chart area and choose **Customize** > **Memory** > **Oozie Direct Buffer Resource Percentage**. Click **OK**.

Step 3 Check whether the used direct memory of Oozie reaches the threshold (the default value is 80% of the maximum direct memory) specified for Oozie.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Oozie** > **Configurations**. Click **All Configurations**. Search **GC_OPTS** in the search box. Increase the value of **-XX:MaxDirectMemorySize** as required, and click **Save**. Click **OK**.

NOTE

Suggestions on GC parameter settings for Oozie:

You are advised to set the value of **-XX:MaxDirectMemorySize** to 1/4 of the value of **-Xmx**. For example, if **-Xmx** is set to 4 GB, **-XX:MaxDirectMemorySize** is set to 1024 MB. If **-Xmx** is set to 2 GB, **-XX:MaxDirectMemorySize** is set to 512 MB. It is recommended that the value of **-XX:MaxDirectMemorySize** be greater than or equal to 512 MB.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **Oozie** in the required cluster from the **Service** drop-down list.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.121 ALM-17007 Garbage Collection (GC) Time of the Oozie Process Exceeds the Threshold

Description

The system checks GC time of the Oozie process every 60 seconds. The alarm is generated when GC time of the Oozie process exceeds the threshold (default value: **12 seconds**). The alarm is cleared when GC time is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
17007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Oozie responds slowly when it is used to submit tasks.

Possible Causes

The heap memory of the Oozie instance is overused or the heap memory is inappropriately allocated.

Procedure

Check GC time.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **Garbage Collection (GC) Time of the Oozie Process Exceeds the Threshold** > **Location**. Check the IP address of the instance involved in this alarm.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Oozie** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down menu in the chart area and choose **Customize** > **GC** > **Garbage Collection (GC) Time of Oozie**. Click **OK**.

Step 3 Check whether GC time of the Oozie process every second exceeds the threshold (default value: **12 seconds**).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* > **Services** > **Oozie** > **Configurations**. Click **All Configurations**. Search **GC_OPTS** in the search box. Increase the value of **-Xmx** as required, and click **Save**. Click **OK**.

NOTE

Suggestions on GC parameter settings for Oozie:

You are advised to set **-Xms** and **-Xmx** to the same value to prevent adverse impact on performance when JVM dynamically adjusts the heap memory size.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **Oozie** in the required cluster from the **Service** drop-down list.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.122 ALM-18000 Yarn Service Unavailable

Description

This alarm is generated when the Yarn service is unavailable. The alarm module checks the Yarn service status every 60 seconds.

The alarm is cleared when the Yarn service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceNam	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The cluster cannot provide Yarn services. Users cannot run new applications. Submitted applications cannot be run.

Possible Causes

- The ZooKeeper service is abnormal.
- The HDFS service is abnormal.
- There is no active ResourceManager instance in the Yarn cluster.
- All the NodeManagers in the Yarn cluster are abnormal.

Procedure

Check ZooKeeper service status.

Step 1 On the FusionInsight Manager, check whether the alarm list contains **ALM-13000 ZooKeeper Service Unavailable**.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Rectify the fault by following the steps provided in **ALM-13000 ZooKeeper Service Unavailable**, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Check the HDFS service status.

Step 3 On the FusionInsight Manager, check whether the alarm list contains the HDFS alarms.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Choose **O&M > Alarm > Alarms**, handle HDFS alarms based on the alarm help, and check whether the Yarn alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check the ResourceManager status in the Yarn cluster.

Step 5 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn**.

Step 6 In **Dashboard**, check whether there is an active ResourceManager instance in the Yarn cluster.

- If yes, go to **Step 7**.
- If no, go to **Step 10**.

Check the NodeManager node status in the Yarn cluster.

Step 7 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance**.

Step 8 Query NodeManager **Running Status**, and check whether there are unhealthy nodes.

- If yes, go to **Step 9**.
- If no, go to **Step 10**.

Step 9 Rectify the fault by following the steps provided in **ALM-18002 NodeManager Heartbeat Lost** or **ALM-18003 NodeManager Unhealthy**. After the fault is rectified, check whether the Yarn alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

Collect fault information.

Step 10 On the FusionInsight Manager portal of the active cluster, choose **O&M** > **Log > Download**.

Step 11 Select **Yarn** in the required cluster from the **Service**.

Step 12 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.123 ALM-18002 NodeManager Heartbeat Lost

Description

The system checks the number of lost NodeManager nodes every 30 seconds, and compares the number with the threshold. The Number of Lost Nodes indicator has a default threshold. The alarm is generated when the value of Number of Lost Nodes exceeds the threshold.

To change the threshold, on FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn**. On the displayed page, choose **Configurations** > **All Configurations**, and change the value of **yarn.nodemanager.lost.alarm.threshold**. You do not need to restart Yarn to make the change take effect.

The default threshold is 0. The alarm is generated when the number of lost nodes exceeds the threshold, and is cleared when the number of lost nodes is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18002	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Lost Host	Specifies the list of hosts with lost nodes.

Impact on the System

- The lost NodeManager node cannot provide the Yarn service.
- The number of containers decreases, so the cluster performance deteriorates.

Possible Causes

- NodeManager is forcibly deleted without decommission.
- All the NodeManager instances are stopped or the NodeManager process is faulty.
- The host where the NodeManager node resides is faulty.
- The network between the NodeManager and ResourceManager is disconnected or busy.

Procedure

Check the NodeManager status.

Step 1 On the FusionInsight Manager, and choose **O&M** > **Alarm > Alarms**. Click ⌄ before the alarm and obtain lost nodes in **Additional Information**.

Step 2 Check whether the lost nodes are hosts that have been manually deleted without decommission.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 After the setting, Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn**. On the displayed page, choose **Configurations** > **All Configurations**. Search for **yarn.nodemanager.lost.alarm.threshold** and change its value to the number of hosts that are not out of service and proactively deleted. After the setting, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Step 4 Manually clear the alarm. Note that decommission must be performed before deleting hosts.

Step 5 On the FusionInsight Manager portal, choose **Cluster > Hosts**, and check whether the nodes obtained in **Step 1** are healthy.

- If yes, go to **Step 7**.
- If no, go to **Step 6**.

Step 6 Rectify the node fault based on **ALM-12006 Node Fault** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Check the process status.

Step 7 On the FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* **> Services** > **Yarn** > **Instance**, and check whether there are NodeManager instances whose status is not **Good**.

- If yes, go to **Step 10**.
- If no, go to **Step 8**.

Step 8 Check whether the NodeManager instance is deleted.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

Step 9 Restart the active and standby ResourceManager instances, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Check the instance status.

Step 10 Select NodeManager instances which running state is not **Normal** and restart them. Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Check the network status.

Step 11 Log in to the management node, **ping** the IP address of the lost NodeManager node to check whether the network is disconnected or busy.

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 Rectify the network, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collect fault information.

Step 13 On the FusionInsight Manager in the active cluster, choose **O&M** > **Log > Download**.

Step 14 Select **Yarn** in the required cluster from the **Service**.

Step 15 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.124 ALM-18003 NodeManager Unhealthy

Description

The system checks the number of unhealthy NodeManager nodes every 30 seconds, and compares the number with the threshold. The Unhealthy Nodes indicator has a default threshold. This alarm is generated when the value of the Unhealthy Nodes indicator exceeds the threshold.

To change the threshold, on FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn**. On the displayed page, choose **Configurations** > **All Configurations**, and change the value of **yarn.nodemanager.unhealthy.alarm.threshold**. You do not need to restart Yarn to make the change take effect.

The default threshold is 0. The alarm is generated when the number of unhealthy nodes exceeds the threshold, and is cleared when the number of unhealthy nodes is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18003	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Unhealthy Host	Specifies the list of hosts with unhealthy nodes.

Impact on the System

- The faulty NodeManager node cannot provide the Yarn service.
- The number of containers decreases, so the cluster performance deteriorates.

Possible Causes

- The hard disk space of the host where the NodeManager node resides is insufficient.
- User **omm** does not have the permission to access a local directory on the NodeManager node.

Procedure

Check the hard disk space of the host.

Step 1 On the FusionInsight Manager, and choose **O&M** > **Alarm > Alarms**. Click before the alarm and obtain unhealthy nodes in **Additional Information**.

Step 2 Choose **Cluster** > *Name of the desired cluster* **> Services > Yarn** > **Instance**, select the NodeManager instance corresponding to the host, choose **Instance Configurations > All Configurations** and view disks corresponding to **yarn.nodemanager.local-dirs** and **yarn.nodemanager.log-dirs**.

Step 3 Choose **O&M > Alarm > Alarms**. In the alarm list, check whether the related disk has the alarm **ALM-12017 Insufficient Disk Capacity**.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Rectify the disk fault based on **ALM-12017 Insufficient Disk Capacit** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Step 5 Choose **Hosts** > *Name of the desired host* . On the **Dashboard** page, check the disk usage of the corresponding partition. Check whether the percentage of the used space of the mounted disk exceeds the value of **yarn.nodemanager.disk-health-checker.max-disk-utilization-per-disk-percentage**

- If yes, go to **Step 6**.

- If no, go to **Step 7**.

Step 6 Reduce the disk usage to less than the value of **yarn.nodemanager.disk-health-checker.max-disk-utilization-per-disk-percentage**, wait for 10 to 20 minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Check the access permission of the local directory on each NodeManager node.

Step 7 Obtain the NodeManager directory viewed in **Step 2**, log in to each NodeManager node as user **root**, and go to the obtained directory.

Step 8 Run the **ll** command to check whether the permission of the **localdir** and **containerlogs** folders is **755** and whether **User:Group** is **omm:ficommon**.

- If yes, no further action is required.
- If no, go to **Step 9**.

Step 9 Run the following command to set the permission to **755** and **User:Group** to **omm:ficommon**:

chmod 755 *<folder_name>*

chown omm:ficommon *<folder_name>*

Step 10 Wait for 10 to 20 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager in the active cluster, choose **O&M** > **Log** > **Download**.

Step 12 Select **Yarn** in the required cluster from the **Service**.

Step 13 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.125 ALM-18008 Heap Memory Usage of ResourceManager Exceeds the Threshold

Description

The system checks the heap memory usage of Yarn ResourceManager every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the heap memory usage of Yarn ResourceManager exceeds the threshold (95% of the maximum memory by default).

Users can choose **O&M > Alarm > Thresholds** > *Name of the desired cluster* > **Yarn** to change the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the heap memory usage of Yarn ResourceManager is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the heap memory usage of Yarn ResourceManager is less than or equal to 95% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the heap memory usage of Yarn ResourceManager is overhigh, the performance of Yarn task submission and operation is affected. In addition, a memory overflow may occur so that the Yarn service is unavailable.

Possible Causes

The heap memory of the Yarn ResourceManager instance on the node is overused or the heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **Heap Memory Usage of Yarn ResourceManager Exceeds the Threshold** > **Location**. Check the HostName of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **ResourceManager** (Indicates the host name of the instance for which the alarm is generated)**.** Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **ResourceManager > Percentage of Used Memory of the ResourceManager**. Check the heap memory usage.

Step 3 Check whether the used heap memory of ResourceManager reaches 95% of the maximum heap memory specified for ResourceManager.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations** > **ResourceManager** > **System**. Increase the value of **GC_OPTS** parameter as required, click **Save**. Restart the role instance.

NOTE

The mapping between the number of NodeManager instances in a cluster and the memory size of ResourceManager is as follows:

- If the number of NodeManager instances in the cluster reaches 100, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 200, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 500, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms10G -Xmx10G -XX:NewSize=1G -XX:MaxNewSize=2G
- If the number of NodeManager instances in the cluster reaches 1000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms20G -Xmx20G -XX:NewSize=1G -XX:MaxNewSize=2G
- If the number of NodeManager instances in the cluster reaches 2000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms40G -Xmx40G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 3000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms60G -Xmx60G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 4000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms80G -Xmx80G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 5000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms100G -Xmx100G -XX:NewSize=3G -XX:MaxNewSize=6G

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select the following node in the required cluster from the **Service**.

- NodeAgent
- Yarn

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.126 ALM-18009 Heap Memory Usage of JobHistoryServer Exceeds the Threshold

Description

The system checks the heap memory usage of Mapreduce JobHistoryServer every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the heap memory usage of Mapreduce JobHistoryServer exceeds the threshold (95% of the maximum memory by default).

Users can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Mapreduce** to change the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the heap memory usage of MapReduce JobHistoryServer is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the heap memory usage of MapReduce JobHistoryServer is less than or equal to 95% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the heap memory usage of Mapreduce JobHistoryServer is overhigh, the performance of Mapreduce log archiving is affected. In addition, a memory overflow may occur so that the Yarn service is unavailable.

Possible Causes

The heap memory of the Mapreduce JobHistoryServer instance on the node is overused or the heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18009 Heap Memory Usage of MapReduce JobHistoryServer Exceeds the Threshold** > **Location**. Check the HostName of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* **> Services** > **Mapreduce** > **Instance** > **JobHistoryServer.** Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **JobHistoryServer heap memory usage statistics**. JobHistoryServer indicates the corresponding HostName of the instance for which the alarm is generated. Check the heap memory usage.

Step 3 Check whether the used heap memory of JobHistoryServer reaches 95% of the maximum heap memory specified for JobHistoryServer.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* **> Services** > **Mapreduce** > **Configurations** > **All Configurations** > **JobHistoryServer** > **System**. Increase the value of **GC_OPTS** parameter as required, click **Save**. Click **OK** and restart the role instance.

NOTE

The mapping between the number of historical tasks (10000) and the memory of JobHistoryServer is as follows:

-Xms30G -Xmx30G -XX:NewSize=1G -XX:MaxNewSize=2G

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 7 Select the following node in the required cluster from the **Service**.

- NodeAgent
- Mapreduce

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.127 ALM-18010 ResourceManager GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the ResourceManager process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default).

This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18010	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the ResourceManager process may interrupt the services.

Possible Causes

The heap memory of the ResourceManager instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18010 ResourceManager GC Time Exceeds the Threshold** > **Location** to check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **ResourceManager (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Garbage Collection (GC) Time of ResourceManager** to check the GC duration statistics of the Broker process collected every minute.

Step 3 Check whether the GC duration of the ResourceManager process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Configurations** > **All Configurations** > **ResourceManager** > **System** to increase the value of **GC_OPTS** parameter as required.

NOTE

The mapping between the number of NodeManager instances in a cluster and the memory size of ResourceManager is as follows:

- If the number of NodeManager instances in the cluster reaches 100, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 200, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 500, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms10G -Xmx10G -XX:NewSize=1G -XX:MaxNewSize=2G
- If the number of NodeManager instances in the cluster reaches 1000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms20G -Xmx20G -XX:NewSize=1G -XX:MaxNewSize=2G
- If the number of NodeManager instances in the cluster reaches 2000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms40G -Xmx40G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 3000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms60G -Xmx60G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 4000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms80G -Xmx80G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 5000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms100G -Xmx100G -XX:NewSize=3G -XX:MaxNewSize=6G

Step 5 Save the configuration and restart the ResourceManager instance.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **ResourceManager** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.128 ALM-18011 NodeManager GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the NodeManager process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default).

This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18011	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the NodeManager process may interrupt the services.

Possible Causes

The heap memory of the NodeManager instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18011 NodeManager GC Time Exceeds the Threshold** > **Location** to check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **NodeManager (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Garbage Collection (GC) Time of NodeManager** to check the GC duration statistics of the Broker process collected every minute.

Step 3 Check whether the GC duration of the NodeManager process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations** > **NodeManager** > **System** to increase the value of **GC_OPTS** parameter as required.

NOTE

The mapping between the number of NodeManager instances in a cluster and the memory size of NodeManager is as follows:

- If the number of NodeManager instances in the cluster reaches 100, the recommended JVM parameters for NodeManager instances are as follows: -Xms2G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 200, the recommended JVM parameters for NodeManager instances are as follows: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 500, the recommended JVM parameters for NodeManager instances are as follows: -Xms8G -Xmx8G -XX:NewSize=1G -XX:MaxNewSize=2G

Step 5 Save the configuration and restart the NodeManager instance.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **NodeManager** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.129 ALM-18012 JobHistoryServer GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the JobHistoryServer process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default).

This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18012	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the JobHistoryServer process may interrupt the services.

Possible Causes

The heap memory of the JobHistoryServer instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18012 JobHistoryServer GC Time Exceeds the Threshold** > **Location** to check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **MapReduce** > **Instance** > **JobHistoryServer (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Garbage Collection (GC) Time of the JobHistoryServer** to check the GC duration statistics of the Broker process collected every minute.

Step 3 Check whether the GC duration of the JobHistoryServer process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **MapReduce** > **Configurations** > **All Configurations** > **JobHistoryServer** > **System** to increase the value of **GC_OPTS** parameter as required.

NOTE

The mapping between the number of historical tasks (10000) and the memory of the JobHistoryServer is as follows:

-Xms30G -Xmx30G -XX:NewSize=1G -XX:MaxNewSize=2G

Step 5 Save the configuration and restart the JobHistoryServer instance.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **JobHistoryServer** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.130 ALM-18013 ResourceManager Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the Yarn service every 30 seconds. This alarm is generated when the direct memory usage of a ResourceManager instance exceeds the threshold (90% of the maximum memory).

The alarm is cleared when the direct memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18013	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the Yarn service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the ResourceManager instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18013 ResourceManager Direct Memory Usage Exceeds the Threshold** > **Location** to check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **ResourceManager (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Memory Usage Status of ResourceManager** to check the direct memory usage.

Step 3 Check whether the used direct memory of ResourceManager reaches 90% of the maximum direct memory specified for ResourceManager by default.

- If yes, go to **Step 4**.
- If no, go to **Step 9**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations** > **ResourceManager** > **System** to increase the value of check whether **-XX:MaxDirectMemorySize** exists in the **GC_OPTS** parameter.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 In the **GC_OPTS** parameter, delete **-XX:MaxDirectMemorySize**.

Step 6 Save the configuration and restart the ResourceManager instance.

Step 7 Check whether the **ALM-18008 Heap Memory Usage of ResourceManager Exceeds the Threshold** exists.

- If yes, handle the alarm by referring to **ALM-18008 Heap Memory Usage of ResourceManager Exceeds the Threshold**.
- If no, go to **Step 8**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **ResourceManager** in the required cluster from the **Service**.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.131 ALM-18014 NodeManager Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the Yarn service every 30 seconds. This alarm is generated when the direct memory usage of a NodeManager instance exceeds the threshold (90% of the maximum memory).

The alarm is cleared when the direct memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18014	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the Yarn service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the NodeManager instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm > Alarms** > **ALM-18014 NodeManager Direct Memory Usage Exceeds the Threshold** > **Location** to check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **NodeManager (IP address for which the alarm is generated)**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Resource** > **Percentage of Used Memory of the NodeManager** to check the direct memory usage.

Step 3 Check whether the used direct memory of NodeManager reaches 90% of the maximum direct memory specified for NodeManager by default.

- If yes, go to **Step 4**.
- If no, go to **Step 9**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations**> **NodeManager** > **System** to check whether "-XX:MaxDirectMemorySize" exists in the **GC_OPTS** parameter.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 In the **GC_OPTS** parameter, delete "-XX:MaxDirectMemorySize".

Step 6 Save the configuration and restart the NodeManager instance.

Step 7 Check whether the **ALM-18018 NodeManager Heap Memory Usage Exceeds the Threshold** exists.

- If yes, handle the alarm by referring to **ALM-18018 NodeManager Heap Memory Usage Exceeds the Threshold**.
- If no, go to **Step 8**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **NodeManager** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.132 ALM-18015 JobHistoryServer Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the MapReduce service every 30 seconds. This alarm is generated when the direct memory usage of a JobHistoryServer instance exceeds the threshold (90% of the maximum memory).

The alarm is cleared when the direct memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18015	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the MapReduce service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the JobHistoryServer instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18015 JobHistoryServer Direct Memory Usage Exceeds the Threshold** > **Location** to check the IP address of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **MapReduce** > **Instance** > **JobHistoryServer (IP address for which the alarm is generated).** Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Memory Usage Status of JobHistoryServer** to check the direct memory usage.

Step 3 Check whether the used direct memory of JobHistoryServer reaches 90% of the maximum direct memory specified for JobHistoryServer by default.

- If yes, go to **Step 4**.
- If no, go to **Step 9**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **MapReduce** > **Configurations** > **All Configurations** > **JobHistoryServer** > **System** to check whether "-XX:MaxDirectMemorySize" exists in the **GC_OPTS** parameter.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 In the **GC_OPTS** parameter, delete "-XX:MaxDirectMemorySize".

Step 6 Save the configuration and restart the JobHistoryServer instance.

Step 7 Check whether the **ALM-18009 Heap Memory Usage of JobHistoryServer Exceeds the Threshold** exists.

- If yes, handle the alarm by referring to **ALM-18009 Heap Memory Usage of JobHistoryServer Exceeds the Threshold**.
- If no, go to **Step 8**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 10 Select **JobHistoryServer** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.133 ALM-18016 Non Heap Memory Usage of ResourceManager Exceeds the Threshold

Description

The system checks the Non Heap memory usage of Yarn ResourceManager every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the Non Heap memory usage of Yarn ResourceManager exceeds the threshold (90% of the maximum memory by default).

Users can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Yarn** to change the threshold.

The alarm is cleared when the Non Heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18016	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.

Name	Meaning
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the Non Heap memory usage of Yarn ResourceManager is overhigh, the performance of Yarn task submission and operation is affected. In addition, a memory overflow may occur so that the Yarn service is unavailable.

Possible Causes

The Non Heap memory of the Yarn ResourceManager instance on the node is overused or the Non Heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the Non Heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18016 Non Heap Memory Usage of Yarn ResourceManager Exceeds the Threshold** > **Location**. Check the HostName of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **ResourceManager**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Percentage of Used Memory of the ResourceManager**. ResourceManager indicates the corresponding HostName of the instance for which the alarm is generated. Check the Non Heap memory usage.

Step 3 Check whether the used Non Heap memory of ResourceManager reaches 90% of the maximum Non Heap memory specified for ResourceManager by default.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations** > **ResourceManager** > **System**. Adjust the **GC_OPTS** memory parameter of ResourceManager. Save the configuration and restart the ResourceManager instance.

NOTE

The mapping between the number of NodeManager instances in a cluster and the memory size of ResourceManager is as follows:

- If the number of NodeManager instances in the cluster reaches 100, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 200, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 500, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms10G -Xmx10G -XX:NewSize=1G -XX:MaxNewSize=2G
- If the number of NodeManager instances in the cluster reaches 1000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms20G -Xmx20G -XX:NewSize=1G -XX:MaxNewSize=2G
- If the number of NodeManager instances in the cluster reaches 2000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms40G -Xmx40G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 3000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms60G -Xmx60G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 4000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms80G -Xmx80G -XX:NewSize=2G -XX:MaxNewSize=4G
- If the number of NodeManager instances in the cluster reaches 5000, the recommended JVM parameters of the ResourceManager instance are as follows: -Xms100G -Xmx100G -XX:NewSize=3G -XX:MaxNewSize=6G

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select the following node in the required cluster from the **Service**.

- NodeAgent
- Yarn

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.134 ALM-18017 Non Heap Memory Usage of NodeManager Exceeds the Threshold

Description

The system checks the Non Heap memory usage of Yarn NodeManager every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the Non Heap memory usage of Yarn NodeManager exceeds the threshold (90% of the maximum memory by default).

Users can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Yarn** to change the threshold.

The alarm is cleared when the Non Heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18017	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the Non Heap memory usage of Yarn NodeManager is overhigh, the performance of Yarn task submission and operation is affected. In addition, a memory overflow may occur so that the Yarn service is unavailable.

Possible Causes

The Non Heap memory of the Yarn NodeManager instance on the node is overused or the Non Heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the Non Heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18017 Non Heap Memory Usage of Yarn NodeManager Exceeds the Threshold** > **Location**. Check the HostName of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **NodeManager**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Resource** > **Percentage of Used Memory of the NodeManager**. NodeManager indicates the corresponding HostName of the instance for which the alarm is generated. Check the Non Heap memory usage.

Step 3 Check whether the used Non Heap memory of NodeManager reaches 90% of the maximum Non Heap memory specified for NodeManager by default.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations** > **NodeManager** > **System**. Adjust the **GC_OPTS** memory parameter of NodeManager, click **Save**, and click **OK,** and restart the role instance.

NOTE

The mapping between the number of NodeManager instances in a cluster and the memory size of NodeManager is as follows:

- If the number of NodeManager instances in the cluster reaches 100, the recommended JVM parameters for NodeManager instances are as follows: -Xms2G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 200, the recommended JVM parameters for NodeManager instances are as follows: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 500, the recommended JVM parameters for NodeManager instances are as follows: -Xms8G -Xmx8G -XX:NewSize=1G -XX:MaxNewSize=2G

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 7 Select the following node in the required cluster from the **Service**.

- NodeAgent
- Yarn

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.135 ALM-18018 NodeManager Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of Yarn NodeManager every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the heap memory usage of Yarn NodeManager exceeds the threshold (95% of the maximum memory by default).

The alarm is cleared when the heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18018	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.

Name	Meaning
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the heap memory usage of Yarn NodeManager is overhigh, the performance of Yarn task submission and operation is affected. In addition, a memory overflow may occur so that the Yarn service is unavailable.

Possible Causes

The heap memory of the Yarn NodeManager instance on the node is overused or the heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18018 NodeManager Heap Memory Usage Exceeds the Threshold** > **Location**. Check the HostName of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Instance** > **NodeManager**. Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **Resource** > **Percentage of Used Memory of the NodeManager** to check the heap memory usage.

Step 3 Check whether the used heap memory of NodeManager reaches 95% of the maximum heap memory specified for NodeManager.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations** > **All Configurations** > **NodeManager** > **System**. Increase the value of **GC_OPTS** parameter as required, click **Save**, and click **OK**, and restart the role instance.

NOTE

The mapping between the number of NodeManager instances in a cluster and the memory size of NodeManager is as follows:

- If the number of NodeManager instances in the cluster reaches 100, the recommended JVM parameters for NodeManager instances are as follows: -Xms2G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 200, the recommended JVM parameters for NodeManager instances are as follows: -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=1G
- If the number of NodeManager instances in the cluster reaches 500, the recommended JVM parameters for NodeManager instances are as follows: -Xms8G -Xmx8G -XX:NewSize=1G -XX:MaxNewSize=2G

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 7 Select the following node in the required cluster from the **Service**.

- NodeAgent
- Yarn

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.136 ALM-18019 Non Heap Memory Usage of JobHistoryServer Exceeds the Threshold

Description

The system checks the Non Heap memory usage of MapReduce JobHistoryServer every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the Non Heap memory usage of MapReduce JobHistoryServer exceeds the threshold (90% of the maximum memory by default).

Users can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **MapReduce** to change the threshold.

The alarm is cleared when the Non Heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18019	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the Non Heap memory usage of MapReduce JobHistoryServer is overhigh, the performance of MapReduce task submission and operation is affected. In addition, a memory overflow may occur so that the MapReduce service is unavailable.

Possible Causes

The Non Heap memory of the MapReduce JobHistoryServer instance on the node is overused or the Non Heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the Non Heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** > **ALM-18019 Non Heap Memory Usage of MapReduce JobHistoryServer Exceeds the Threshold** > **Location**. Check the HostName of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **MapReduce** > **Instance** > **JobHistoryServer.** Click the drop-down menu in the upper right corner of **Chart**, choose **Customize** > **JobHistoryServer Non Heap memory usage statistics**. JobHistoryServer indicates the corresponding HostName of the instance for which the alarm is generated. Check the Non Heap memory usage.

Step 3 Check whether the used Non Heap memory of JobHistoryServer reaches 90% of the maximum Non Heap memory specified for JobHistoryServer.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **MapReduce** > **Configurations** > **All Configurations** > **JobHistoryServer** > **System**. Adjust the **GC_OPTS** memory parameter of the NodeManager, click **Save**, and click **OK,** and restart the role instance.

NOTE

The mapping between the number of historical tasks (10000) and the memory of the JobHistoryServer is as follows:

-Xms30G -Xmx30G -XX:NewSize=1G -XX:MaxNewSize=2G

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 7 Select the following node in the required cluster from the **Service**.

- NodeAgent
- MapReduce

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.137 ALM-18020 Yarn Task Execution Timeout

Description

The system checks MapReduce and Spark tasks (except for permanent JDBC tasks) submitted to Yarn every 15 minutes. This alarm is generated when the task execution time exceeds the timeout duration specified by the user. However, the task can be properly executed. The client timeout parameter of MapReduce is mapreduce.application.timeout.alarm and that of Spark is spark.application.timeout.alarm. The unit is ms.

This alarm is cleared when the task is finished or terminated.

Attribute

Alarm ID	Alarm Severity	Auto Clear
18020	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
ApplicationName	Specifies the object (application ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

The alarm persists after task execution times out. However, the task can still be properly executed, so this alarm does not exert any impact on the system.

Possible Causes

- The specified timeout duration is shorter than the required execution time.
- The queue resources for task running are insufficient.
- Task data skew occurs. As a result, some tasks process a large amount of data and take a long time to execute.

Procedure

Check whether the timeout interval is correctly set.

Step 1 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. The **Alarms** page is displayed.

Step 2 Select the alarm whose ID is **18020**. In the alarm details, view **Location** to obtain the timeout task name and timeout duration.

Step 3 Based on the task name and timeout interval, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **ResourceManager (Active)** to log in to the native Yarn page. Then find the task on the native page, check its **StartTime** and calculate the task execution time based on the current system time. Check whether the task execution time exceeds the timeout duration.

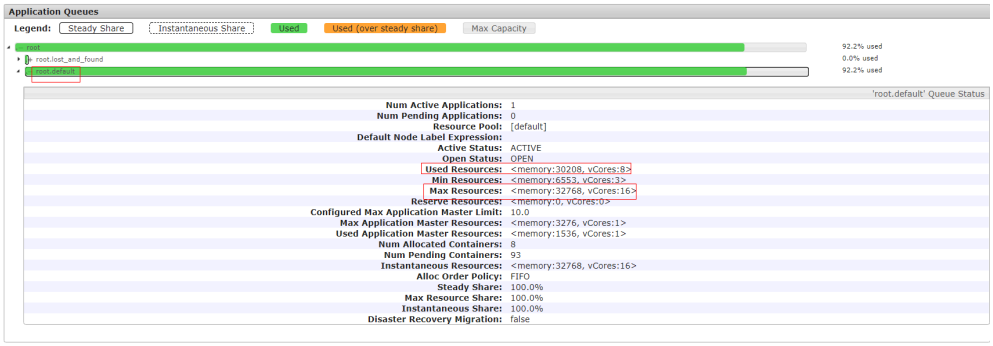
- If yes, go to **Step 5**.
- If no, go to **Step 10**.

Step 4 Evaluate the expected task execution time based on the service and compare it with the task timeout interval. If the timeout interval is too short, set the timeout interval (**mapreduce.application.timeout.alarm** or **spark.application.timeout.alarm**) of the client to the task expected execution time. Run the task again and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the queue resources are sufficient.

Step 5 Find the task on the native page and view the queue name of the task. Click **Scheduler** on the left of the native page. On the **Applications Queues** page, find the corresponding queue name and expand the queue details, as shown in the following figure.

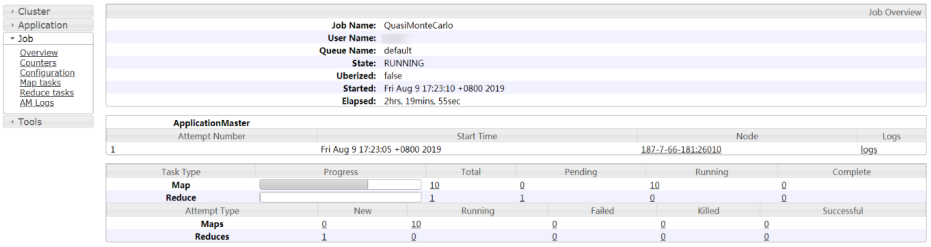


Step 6 Check whether the value of **Used Resources** in the queue details is approximately equal to the value of **Max Resources**, which indicates that the resources in the queue submitted by the task have been used up. If the queue resources are insufficient, choose **Tenant Resources** > **Dynamic Resource Plan** > **Resource Distribution Policy** on FusionInsight Manager and increase the value of **Max Resources** for the queue. Run the task again and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Check whethercoccurs.

Step 7 On the native Yarn page, click *task ID* (for example, **application_1565337919723_0002**) > **Tracking URL:ApplicationMaster** > **job_1565337919723_0002**. The following page is displayed.



Step 8 Choose **Job** > **Map tasks** or **Job** > **Reduce tasks** on the left and check whether the execution time of each Map or Reduce task differs greatly. If yes, task data skew occurs. In this case, you need to balance the task data.

Step 9 Rectify the fault based on the preceding causes and perform the tasks again. Then, check whether the alarm persists.

- If yes, go to **Step 10**.
- If no, no further action is required.

Collect the fault information.

Step 10 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 11 Expand the **Service** drop-down list, and select **Yarn** for the target cluster.

Step 12 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.138 ALM-18021 Mapreduce Service Unavailable

Description

The alarm module checks the MapReduce service status every 60 seconds. This alarm is generated when the system detects that the MapReduce service is unavailable.

The alarm is cleared when the MapReduce service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
18021	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The cluster cannot provide the MapReduce service. For example, MapReduce cannot be used to view task logs or the log archive function is unavailable.

Possible Causes

- The JobHistoryServer instance is abnormal.
- The KrbServer service is abnormal.
- The ZooKeeper service abnormal.
- The HDFS service abnormal.
- The Yarn service is abnormal.

Procedure

Check MapReduce service JobHistoryServer instance status.

Step 1 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **MapReduce** > **Instance**.

Step 2 Check whether the running status of JobHistoryServer is **Normal**.

- If yes, go to **Step 11**.
- If no, go to **Step 3**.

Check the KrbServer service status.

Step 3 In the alarm list on FusionInsight Manager, check whether **ALM-25500 KrbServer Service Unavailable** exists.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Rectify the fault by following the steps provided in **ALM-25500 KrbServer Service Unavailable**, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check the ZooKeeper service.

Step 5 In the alarm list on FusionInsight Manager, check whether **ALM-13000 ZooKeeper Service Unavailable** exists.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 6 Rectify the fault by following the steps provided in **ALM-13000 ZooKeeper Service Unavailable**, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Check the HDFS service status.

Step 7 In the alarm list on FusionInsight Manager, check whether **ALM-14000 HDFS Service Unavailable** exists.

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Rectify the fault by following the steps provided in **ALM-14000 HDFS Service Unavailable**, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Check the Yarn service status.

Step 9 In the alarm list on FusionInsight Manager, check whether **ALM-18000 Yarn Service Unavailable** exists.

- If yes, go to **Step 10**
- If no, go to **Step 11**.

Step 10 Rectify the fault by following the steps provided in **ALM-18000 Yarn Service Unavailable**, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect fault information.

Step 11 On the FusionInsight Manager home page of the active cluster, choose **O&M** > **Log > Download.**

Step 12 Select **MapReduce** in the required cluster from the **Service.**

Step 13 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.139 ALM-18022 Insufficient Yarn Queue Resources

Description

The alarm module checks Yarn queue resources every 60 seconds. This alarm is generated when available resources or ApplicationMaster (AM) resources of a queue are insufficient.

This alarm is cleared when available resources are sufficient.

Attribute

Alarm ID	Alarm Severity	Auto Clear
18022	Minor	Yes

Parameters

Parameter Name	Description
Source	Specifies the cluster for which the alarm is generated.
QueueName	Specifies the queue for which the alarm is generated.
QueueMetric	Specifies the metric of the queue for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

- An application being executed takes longer time.
- An application fails to be executed for a long time after being submitted.

Possible Causes

- NodeManager node resources are insufficient.
- The configured maximum resource capacity of the queue is excessively small.
- The configured maximum AM resource percentage is excessively small.

Procedure

View alarm details.

Step 1 On the FusionInsight Manager, choose **O&M** > **Alarm > Alarms**.

Step 2 View location information of this alarm and check whether **QueueName** is **root** and **QueueMetric** is **Memory** or **QueueName** is **root** and **QueueMetric** is **vCores**.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 The memory or CPU of the Yarn cluster is insufficient. In this case, log in to the node where NodeManager resides and run the **free -g** and **cat /proc/cpuinfo** commands to query the available memory and available CPU of the node, respectively. On FusionInsight Manager, increase the values of **yarn.nodemanager.resource.memory-mb** and **yarn.nodemanager.resource.cpu-vcores** for the Yarn NodeManager based on the query results. Then, restart the NodeManager instance. Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Step 4 If **QueueName** is **<*Tenant Queue*>** and **QueueMetric** is **Memory**, or **QueueName** is **<*Tenant Queue*>** and **QueueMetric** is **vCores** in **Location**, check whether **available Memory =** or **available vCores =** are included in **Additional Information**.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 The memory or CPU of the tenant queue is insufficient. In this case, choose **Tenant Resources** > **Dynamic Resource Plan > Resource Distribution Policy** and increase the value of **Maximum Capacity**. Then, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Step 6 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Configurations** > **All Configurations**. Enter the keyword "threshold" and click **ResourceManager**. Adjust the threshold values of the following parameters:

If **Additional Information** contains **available Memory =**, change the value of **yarn.queue.memory.alarm.threshold** to a value smaller than that of **available Memory =** in **Additional Information**.

If **Additional Information** contains **available vCores =**, change the value of **yarn.queue.vcore.alarm.threshold** to a value smaller than that of **available vCores =** in **Additional Information**.

Wait for five minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Step 7 If **available AmMemory =** or **available AmvCores =** is included in **Additional Information**, ApplicationMaster memory or CPU of the tenant queue is insufficient. In this case, choose **Tenant Resources** > **Dynamic Resource Plan** > **Queue Configuration** and increase the value of **Maximum Am Resource Percent**. Then, check whether this alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Step 8 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Configurations** > **All Configurations**. Enter the keyword "threshold" and click **ResourceManager**. Adjust the threshold values of the following parameters:

If **Additional Information** contains **available AmMemory =**, change the value of **yarn.queue.memory.alarm.threshold** to a value smaller than that of **available AmMemory =** in **Additional Information**.

If **Additional Information** contains **available AmvCores =**, change the value of **yarn.queue.vcore.alarm.threshold** to a value smaller than that of **available AmvCores =** in **Additional Information**.

Wait for five minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 Log in to FusionInsight Manager of the active cluster, and choose **O&M** > **Log** > **Download**.

Step 10 Select **Yarn** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Reference

None

10.13.140 ALM-18023 Number of Pending Yarn Tasks Exceeds the Threshold

Description

The alarm module checks the number of pending applications in the Yarn root queue every 60 seconds. The alarm is generated when the number exceeds 60.

Attribute

Alarm ID	Alarm Severity	Auto Clear
18023	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
QueueName	Identifies the queue for which the alarm is generated.
QueueMetric	Identifies the queue indicator for which the alarm is generated.

Impact on the System

- It takes long time to end an application.
- A new application cannot run after submission.

Possible Causes

- NodeManager node resources are insufficient.
- The maximum resource capacity of the queue and the maximum AM resource percentage are too small.
- The monitoring threshold is too small.

Procedure

Check NodeManager resources.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **ResourceManager (Active)** to access the ResourceManager web UI.

Step 2 Click **Scheduler** and check whether the root queue resources are used up in **Application Queues**.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Expand the capacity of the NodeManager instance of the Yarn service. After the capacity expansion, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check the maximum queue resource capacity and the maximum AM resource percentage.

Step 4 Check whether the resources of the queue corresponding to the pending task are used up.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On FusionInsight Manager, choose **Tenant Resources** > **Dynamic Resource Plan** and add resources as required. Check whether the alarms are cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Adjust the monitoring thresholds.

Step 6 On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Yarn** > **Applications** > **Pending Applications**, and increase the thresholds as required.

Step 7 Check whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect the fault information.

Step 8 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 9 Expand the **Service** drop-down list, and select **Yarn** for the target cluster.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.141 ALM-18024 Pending Yarn Memory Usage Exceeds the Threshold

Description

The alarm module checks the pending memory of Yarn every 60 seconds. The alarm is generated when the pending memory exceeds the threshold. Pending memory indicates the total memory that is not allocated to submitted Yarn applications.

Attribute

Alarm ID	Alarm Severity	Auto Clear
18024	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
QueueName	Identifies the queue for which the alarm is generated.
QueueMetric	Identifies the queue indicator for which the alarm is generated.

Impact on the System

- It takes long time to end an application.
- A new application cannot run after submission.

Possible Causes

- NodeManager node resources are insufficient.
- The maximum resource capacity of the queue and the maximum AM resource percentage are too small.
- The monitoring threshold is too small.

Procedure

Check NodeManager resources.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **ResourceManager (Active)** to access the ResourceManager web UI.

Step 2 Click **Scheduler** and check whether the root queue resources are used up in **Application Queues**.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Expand the capacity of the NodeManager instance of the Yarn service. After the capacity expansion, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check the maximum queue resource capacity and the maximum AM resource percentage.

Step 4 Check whether the resources of the queue corresponding to the pending task are used up.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On FusionInsight Manager, choose **Tenant Resources** > **Dynamic Resource Plan** and add resources as required. Check whether the alarms are cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Adjust the monitoring thresholds.

Step 6 On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Yarn** > **CPU and Memory** > **Pending Memory**, and increase the threshold as required.

Step 7 Check whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect the fault information.

Step 8 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 9 Expand the **Service** drop-down list, and select **Yarn** for the target cluster.

Step 10 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.142 ALM-18025 Number of Terminated Yarn Tasks Exceeds the Threshold

Description

The alarm module checks the number of terminated applications in the Yarn root queue every 60 seconds. The alarm is generated when the number exceeds 50 for three consecutive times.

Attribute

Alarm ID	Alarm Severity	Auto Clear
18025	Major	Yes

Parameters

Name	Meaning
Cluster Name	Specifies the cluster for which the alarm is generated.
Service Name	Specifies the service for which the alarm is generated.
Role Name	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A large number of application tasks are forcibly terminated.

Possible Causes

- The user forcibly terminates a large number of tasks.
- The system terminates tasks due to some error.

Procedure

Check the alarm details.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** to go to the alarm page.

Step 2 View **Additional Information** in the alarm details to check whether the alarm threshold is too small.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Yarn** > **Other** > **Terminated Applications of root queue** to modify the threshold. Go to **Step 6**.

Step 4 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **ResourceManager(Active)** to access the ResourceManager web UI.

Step 5 Click **KILLED** in **Applications** and click the task on the top. View the description of **Diagnostics** and rectify the fault based on the task termination details (for example, the task is terminated by a user).

Step 6 Wait for 3 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect the fault information.

Step 7 On the FusionInsight Manager, choose **O&M > Log > Download**.

Step 8 Expand the **Service** drop-down list, and select **Yarn** for the target cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.143 ALM-18026 Number of Failed Yarn Tasks Exceeds the Threshold

Description

The alarm module checks the number of failed applications in the Yarn root queue every 60 seconds. The alarm is generated when the number exceeds 50 for three consecutive times.

Attribute

Alarm ID	Alarm Severity	Auto Clear
18026	Major	Yes

Parameters

Name	Meaning
Cluster Name	Specifies the cluster for which the alarm is generated.
Service Name	Specifies the service for which the alarm is generated.
Role Name	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

- A large number of application tasks fail to be executed.
- Failed tasks need to be submitted again.

Possible Causes

The task fails to be executed due to some error.

Procedure

Check the alarm details.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** to go to the alarm page.

Step 2 View **Additional Information** in the alarm details to check whether the alarm threshold is too small.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Yarn** > **Other** > **Failed Applications of root queue** to modify the threshold. Go to **Step 6**.

Step 4 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **ResourceManager(Active)** to access the ResourceManager web UI.

Step 5 Click **FAILED** in **Applications** and click the task on the top. View the description of **Diagnostics** and rectify the fault based on the task failure causes.

Step 6 Wait for 3 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect the fault information.

Step 7 On the FusionInsight Manager, choose O&M > Log > Download.

Step 8 Expand the **Service** drop-down list, and select **Yarn** for the target cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.144 ALM-19000 HBase Service Unavailable

Description

This alarm is generated when the HBase service is unavailable. The alarm module checks the HBase service status every 120 seconds.

This alarm is cleared when the HBase service recovers.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, ServiceName=HBase1 is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Operations, such as reading or writing data and creating tables, cannot be performed.

Possible Causes

- The ZooKeeper service is abnormal.
- The HDFS service is abnormal.
- The HBase service is abnormal.
- The network is abnormal.

Procedure

Check the ZooKeeper service status.

Step 1 On the FusionInsight Manager, check whether the running status of ZooKeeper is **Normal** on service list.

- If yes, go to **Step 5**.
- If no, go to **Step 2**.

Step 2 In the alarm list, check whether **ALM-13000 ZooKeeper Service Unavailable** exists.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Rectify the fault by following the steps provided in **ALM-13000 ZooKeeper Service Unavailable**.

Step 4 Wait several minutes, and check whether alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check the HDFS service status.

Step 5 In the alarm list, check whether **ALM-14000 HDFS Service Unavailable** exists.

- If yes, go to **Step 6**.
- If no, go to **Step 8**.

Step 6 Rectify the fault by following the steps provided in **ALM-14000 HDFS Service Unavailable**.

Step 7 Wait several minutes, and check whether alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Step 8 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**. Check whether **Safe Mode** is **ON**.

- If yes, go to **Step 9**.
- If no, go to **Step 12**.

Step 9 Log in to the HDFS client as user **root**. Run **cd** to switch to the client installation directory, and run **source bigdata_env**.

If the cluster uses the security mode, perform security authentication. Obtain the password of user hdfs from the administrator, run the **kinit hdfs** command and enter the password as prompted.

Step 10 Run the following command to manually exit the safe mode:

hdfs dfsadmin -safemode leave

Step 11 Wait several minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Check the HBase service status.

Step 12 On the FusionInsight Manager portal, click **Cluster** > *Name of the desired cluster* > **Services** > **HBase**.

Step 13 Check whether there is one active HMaster and one standby HMaster.

- If yes, go to **Step 15**.
- If no, go to **Step 14**.

Step 14 Click **Instances**, select the HMaster whose status is not **Active**, click **More**, and select **Restart Instance** to restart the HMaster. Check whether there is one active HMaster and one standby HMaster again.

- If yes, go to **Step 15**.
- If no, go to **Step 21**.

Step 15 Choose **Cluster** >*Name of the desired cluster* > **Services** > **HBase** > **HMaster(Active)** to go to the HMaster WebUI.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

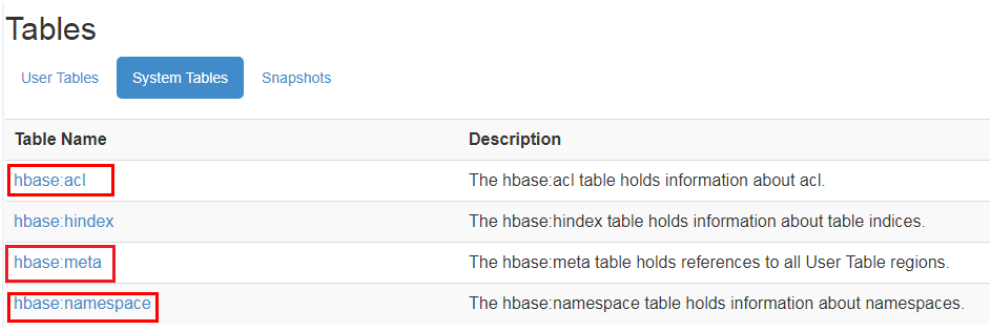
Step 16 Check whether at least one RegionServer exists under **Region Servers**.

- If yes, go to **Step 17**.
- If no, go to **Step 21**.

Step 17 Check **Tables** > **System Tables**, as shown in **Figure 10-34**. Check whether **hbase:meta**, **hbase:namespace**, and **hbase:acl** exist in the **Table Name** column.

- If yes, go to **Step 18**.
- If no, go to **Step 19**.

Figure 10-34 HBase system table



Step 18 As shown in **Figure 10-34**, click the **hbase:meta**, **hbase:namespace**, and **hbase:acl** hyperlinks and check whether the pages are properly displayed. If the pages are properly displayed, the tables are normal.

If they are, go to **Step 19**.

If they are not, go to **Step 23**.

NOTE

In normal mode, **ACL** is enabled for HBase by default. The **hbase:acl** table is generated only when **ACL** is manually enabled. In this case, check this table. In other scenarios, this table does not need to be checked.

Step 19 View the HMaster startup status.

In **Figure 10-35**, if the **RUNNING** state exists in **Tasks**, HMaster is being started. In the **State** column, you can view the time when HMaster is in the **RUNNING** state. In **Figure 10-36**, if the state is **COMPLETE**, HMaster is started.

Check whether HMaster is in the **RUNNING** state for a long time.

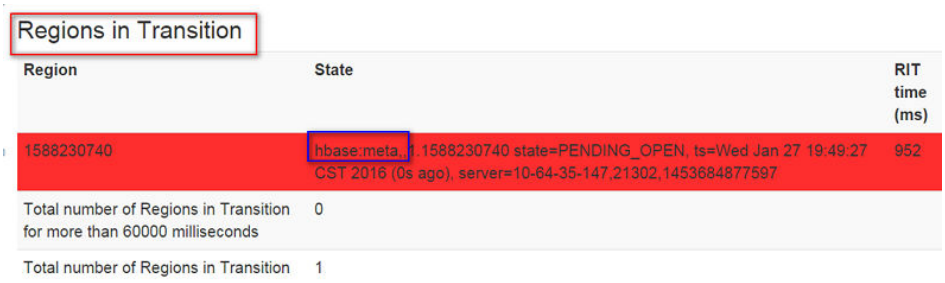
Figure 10-35 HMaster is being started

Figure 10-36 HMaster is started

- If yes, go to **Step 20**.
- If no, go to **Step 21**.

Step 20 On the HMaster WebUI, check whether any hbase:meta is in the **Region in Transition** state for a long time.

Figure 10-37 Region in Transition



- If yes, go to **Step 21**.
- If no, go to **Step 22**.

Step 21 In the precondition that services are not affected, log in to the FusionInsight Manager portal and choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **More** > **Restart Service**. Enter the administrator password and click **OK**.

- If yes, go to **Step 22**.
- If no, go to **Step 23**.

Step 22 Wait several minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 23**.

Check the network connection between HMaster and dependent components.

Step 23 On the FusionInsight Manager, choose **Cluster** >*Name of the desired cluster* > **Services** > **HBase**.

Step 24 Click **Instance** and the HMaster instance list is displayed. Record the **management IP Address** in the row of **HMaster(Active)**.

Step 25 Use the IP address obtained in **Step 24** to log in to the host where the active HMaster runs as user **omm** .

Step 26 Run the **ping** command to check whether communication between the host that runs the active HMaster and the hosts that run the dependent components. (The dependent components include ZooKeeper, HDFS and Yarn. Obtain the IP addresses of the hosts that run these services in the same way as that for obtaining the IP address of the active HMaster.)

- If yes, go to **Step 29**.
- If no, go to **Step 27**.

Step 27 Contact the administrator to restore the network.

Step 28 In the alarm list, check whether **HBase Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 29**.

Collect fault information.

Step 29 On the FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 30 Select the following nodes in the required cluster from the **Service** drop-down list:

- ZooKeeper
- HDFS
- HBase

Step 31 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 32 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.145 ALM-19006 HBase Replication Sync Failed

Description

The alarm module checks the HBase DR data synchronization status every 30 seconds. When disaster recovery (DR) data fails to be synchronized to a standby cluster, the alarm is triggered.

When DR data synchronization succeeds, the alarm is cleared.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19006	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

HBase data in a cluster fails to be synchronized to the standby cluster, causing data inconsistency between active and standby clusters.

Possible Causes

- The HBase service on the standby cluster is abnormal.
- A network exception occurs.

Procedure

Observe whether the system automatically clears the alarm.

Step 1 On the FusionInsight Manager portal of the active cluster, click **O&M** > **Alarm** > **Alarms.**

Step 2 In the alarm list, click the alarm to obtain alarm generation time from **Generated** of the alarm. Check whether the alarm has existed for five minutes.

- If yes, go to **Step 4**.
- If no, go to **Step 3**.

Step 3 Wait five minutes and check whether the system automatically clears the alarm.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check the HBase service status of the standby cluster.

Step 4 Log in to the FusionInsight Manager portal of the active cluster, and click **O&M** > **Alarm** > **Alarms.**

Step 5 In the alarm list, click the alarm to obtain **HostName** from **Location**.

Step 6 Access the node where the HBase client of the active cluster resides as user **omm**.

If the cluster uses a security mode, perform security authentication first and then access the **hbase shell** interface as user **hbase**.

cd /opt/Bigdata/client

source ./bigdata_env

kinit *hbaseuser*

Step 7 Run the **status 'replication', 'source'** command to check the DR synchronization status of the faulty node.

The DR synchronization status of a node is as follows.

```
10-10-10-153:
SOURCE: PeerID=abc, SizeOfLogQueue=0, ShippedBatches=2, ShippedOps=2, ShippedBytes=320,
LogReadInBytes=1636, LogEditsRead=5, LogEditsFiltered=3, SizeOfLogToReplicate=0,
TimeForLogToReplicate=0, ShippedHFiles=0, SizeOfHFileRefsQueue=0, AgeOfLastShippedOp=0,
TimeStampsOfLastShippedOp=Mon Jul 18 09:53:28 CST 2016, Replication Lag=0,
FailedReplicationAttempts=0
SOURCE: PeerID=abc1, SizeOfLogQueue=0, ShippedBatches=1, ShippedOps=1, ShippedBytes=160,
LogReadInBytes=1636, LogEditsRead=5, LogEditsFiltered=3, SizeOfLogToReplicate=0,
TimeForLogToReplicate=0, ShippedHFiles=0, SizeOfHFileRefsQueue=0, AgeOfLastShippedOp=16788,
TimeStampsOfLastShippedOp=Sat Jul 16 13:19:00 CST 2016, Replication Lag=16788,
FailedReplicationAttempts=5
```

Step 8 Obtain **PeerID** corresponding to a record whose **FailedReplicationAttempts** value is greater than 0.

In the preceding step, data on the faulty node 10-10-10-153 fails to be synchronized to a standby cluster whose **PeerID** is **abc1**.

Step 9 Run the **list_peers** command to find the cluster and the HBase instance corresponding to the **PeerID** value.

```
PEER_ID CLUSTER_KEY STATE TABLE_CFS
abc1 10.10.10.110,10.10.10.119,10.10.10.133:2181:/hbase2 ENABLED
abc 10.10.10.110,10.10.10.119,10.10.10.133:2181:/hbase ENABLED
```

In the preceding information, **/hbase2** indicates that data is synchronized to the HBase2 instance of the standby cluster.

Step 10 In the service list of FusionInsight Manager of the standby cluster, check whether the running status of the HBase instance obtained by using **Step 9** is **Normal**.

- If yes, go to **Step 14**.
- If no, go to **Step 11**.

Step 11 In the alarm list, check whether the **ALM-19000 HBase Service Unavailable** alarm is generated.

- If yes, go to **Step 12**.
- If no, go to **Step 14**.

Step 12 Follow troubleshooting procedures in **ALM-19000 HBase Service Unavailable** to rectify the fault.

Step 13 Wait for a few minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 14**.

Check network connections between RegionServers on active and standby clusters.

Step 14 Log in to the FusionInsight Manager portal of the active cluster, and click **O&M** > **Alarm** > **Alarms.**

Step 15 In the alarm list, click the alarm to obtain **HostName** from **Location**.

Step 16 Use the IP address obtained in **Step 15** to log in to a faulty RegionServer node as user **omm**.

Step 17 Run the **ping** command to check whether network connections between the faulty RegionServer node and the host where RegionServer of the standby cluster resides are in the normal state.

- If yes, go to **Step 20**.
- If no, go to **Step 18**.

Step 18 Contact the network administrator to restore the network.

Step 19 After the network is running properly, check whether the alarm is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 20**.

Collect fault information.

Step 20 On the FusionInsight Manager interface of active and standby clusters, choose **O&M** > **Log** > **Download**.

Step 21 In the **Service** drop-down list box, select **HBase** in the required cluster.

Step 22 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 23 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.146 ALM-19007 HBase GC Time Exceeds the Threshold

Description

The system checks the old generation garbage collection (GC) time of the HBase service every 60 seconds. This alarm is generated when the detected old generation GC time exceeds the threshold (exceeds 5 seconds for three consecutive checks by default). To change the threshold, on the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **HBase > GC > GC time for old generation**. This alarm is cleared when the old generation GC time of the HBase service is shorter than or equal to the threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

If the old generation GC time exceeds the threshold, HBase data read and write are affected.

Possible Causes

The memory of HBase instances is overused, the heap memory is inappropriately allocated, or a large number of I/O operations exist in HBase. As a result, GCs occur frequently.

Procedure

Check the GC time.

Step 1 On the FusionInsight Manager portal, click **O&M** > **Alarm** > **Alarms** and select the alarm whose **ID** is **19007**. Then check the role name in **Location** and confirm the IP adress of the instance.

- If the role for which the alarm is generated is HMaster, go to **Step 2**.
- If the role for which the alarm is generated is RegionServer, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance** and click the HMaster for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **GC** > **Garbage Collection (GC) Time of HMaster** and click **OK** to check whether the value of **GC time for old generation** is greater than the threshold (exceeds 5 seconds for three consecutive checks periods by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance** and click the RegionServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **GC** > **Garbage Collection (GC) Time of RegionServer** and click **OK** to check whether the value of **GC time for old generation** is greater than the threshold (exceeds 5 seconds for three consecutive checks periods by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Check the current JVM configuration.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations**, and click **All Configurations**. In Search, enter **GC_OPTS** to check the **GC_OPTS** memory parameter of role HMaster(HBase->HMaster), RegionServer(HBase->RegionServer). Adjust the values of **-Xmx** and **-XX:CMSInitiatingOccupancyFraction** of the GC_OPTS parameter by referring to the Note.

NOTE

1. Suggestions on GC parameter configurations for HMaster
   - Set **-Xms** and **-Xmx** to the same value to prevent JVM from dynamically adjusting the heap memory size and affecting performance.
   - Set **-XX:NewSize** to the value of **-XX:MaxNewSize**, which is one eighth of **-Xmx**.
   - For large-scale HBase clusters with a large number of regions, increase values of **GC_OPTS** parameters for HMaster. Specifically, set **-Xmx** to 4 GB if the number of regions is less than 100,000. If the number of regions is more than 100,000, set -Xmx to be greater than or equal to 6 GB. For each increased 35,000 regions, increase the value of **-Xmx** by 2 GB. The maximum value of **-Xmx** is 32 GB.
2. Suggestions on GC parameter configurations for RegionServer
   - Set **-Xms** and **-Xmx** to the same value to prevent JVM from dynamically adjusting the heap memory size and affecting performance.
   - Set **-XX:NewSize** to one eighth of **-Xmx**.
   - Set the memory for RegionServer to be greater than that for HMaster. If sufficient memory is available, increase the heap memory.
   - Set **-Xmx** based on the machine memory size. Specifically, set **-Xmx** to 32 GB if the machine memory is greater than 200 GB, to 16 GB if the machine memory is greater than 128 GB and less than 200 GB, and to 8 GB if the machine memory is less than 128 GB. When **-Xmx** is set to 32 GB, a RegionServer node supports 2000 regions and 200 hotspot regions.
   - **XX:CMSInitiatingOccupancyFraction** to be less than and equal to **85**, and it is calculated as follows: 100 x (hfile.block.cache.size + hbase.regionserver.global.memstore.size)

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager interface of active and standby clusters, choose **O&M** > **Log** > **Download**.

Step 7 In the **Service** drop-down list box, select **HBase** in the required cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.147 ALM-19008 Heap Memory Usage of the HBase Process Exceeds the Threshold

Description

The system checks the HBase service status every 30 seconds. The alarm is generated when the heap memory usage of an HBase service exceeds the threshold (90% of the maximum memory).

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

If the available HBase heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The heap memory of the HBase service is overused or the heap memory is inappropriately allocated.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, click **O&M** > **Alarm** > **Alarms** and select the alarm whose **ID** is **19008**. Then check the role name in **Location** and confirm the IP adress of the instance.

- If the role for which the alarm is generated is HMaster, go to **Step 2**.
- If the role for which the alarm is generated is RegionServer, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance** and click the HMaster for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory** > **HMaster Heap Memory Usage and Direct Memory Usage Statistics** and click **OK**, check whether the used heap memory of the HBase service reaches 90% of the maximum heap memory specified for HBase.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance** and click the RegionServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory** > **RegionServer Heap Memory Usage and Direct Memory Usage Statistics** and click **OK**, check whether the used heap memory of the HBase service reaches 90% of the maximum heap memory specified for HBase.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations**, and click **All Configurations**. Choose **HMaster/RegionServer** > **System**. Increase the value of **-Xmx** in **GC_OPTS** by referring to the Note.

NOTE

1. Suggestions on GC parameter configurations for HMaster
   - Set **-Xms** and **-Xmx** to the same value to prevent JVM from dynamically adjusting the heap memory size and affecting performance.
   - Set **-XX:NewSize** to the value of **-XX:MaxNewSize**, which is one eighth of **-Xmx**.
   - For large-scale HBase clusters with a large number of regions, increase values of **GC_OPTS** parameters for HMaster. Specifically, set **-Xmx** to 4 GB if the number of regions is less than 100,000. If the number of regions is more than 100,000, set -Xmx to be greater than or equal to 6 GB. For each increased 35,000 regions, increase the value of **-Xmx** by 2 GB. The maximum value of **-Xmx** is 32 GB.
2. Suggestions on GC parameter configurations for RegionServer
   - Set **-Xms** and **-Xmx** to the same value to prevent JVM from dynamically adjusting the heap memory size and affecting performance.
   - Set **-XX:NewSize** to one eighth of **-Xmx**.
   - Set the memory for RegionServer to be greater than that for HMaster. If sufficient memory is available, increase the heap memory.
   - Set **-Xmx** based on the machine memory size. Specifically, set **-Xmx** to 32 GB if the machine memory is greater than 200 GB, to 16 GB if the machine memory is greater than 128 GB and less than 200 GB, and to 8 GB if the machine memory is less than 128 GB. When **-Xmx** is set to 32 GB, a RegionServer node supports 2000 regions and 200 hotspot regions.

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **HBase** in the required cluster from the **Service** drop-down list.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.148 ALM-19009 Direct Memory Usage of the HBase Process Exceeds the Threshold

Description

The system checks the HBase service status every 30 seconds. The alarm is generated when the direct memory usage of an HBase service exceeds the threshold (90% of the maximum memory).

The alarm is cleared when the direct memory usage is less than the threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

If the available HBase direct memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the HBase service is overused or the direct memory is inappropriately allocated.

Procedure

Check direct memory usage.

Step 1 On the FusionInsight Manager portal, click **O&M** > **Alarm** > **Alarms** and select the alarm whose **ID** is **19009**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

- If the role for which the alarm is generated is HMaster, go to **Step 2**.
- If the role for which the alarm is generated is RegionServer, go to **Step 3**.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance** and click the HMaster for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory** > **HMaster Heap Memory Usage and Direct Memory Usage Statistics** and click **OK** to check whether the used direct memory of the HBase service reaches 90% of the maximum direct memory specified for HBase.

- If yes, go to **Step 4**.
- If no, go to **Step 8**.

Step 3 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance** and click the RegionServer for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the **Chart** area and choose **Customize** > **CPU and Memory** > **RegionServer Heap Memory Usage and Direct Memory Usage Statistics** and click **OK** to check whether the used direct memory of the HBase service reaches 90% of the maximum direct memory specified for HBase.

- If yes, go to **Step 4**.
- If no, go to **Step 8**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations**, and click **All Configurations**. Choose **HMaster/RegionServer** > **System** and check whether **XX:MaxDirectMemorySize** exists in **GC_OPTS**.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On the FusionInsight Manager portal, choose **Cluster** > *Nameof the desired cluster* > **Services** > **HBase** > **Configurations**, and click **All Configurations**. Choose **HMaster/RegionServer** > **System** and delete **XX:MaxDirectMemorySize** from **GC_OPTS**.

Step 6 Check whether the **ALM-19008 Heap Memory Usage of the HBase Process Exceeds the Threshold** alarm is generated.

If yes, handle the alarm by referring to **ALM-19008 Heap Memory Usage of the HBase Process Exceeds the Threshold**.

If no, go to **Step 8**.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager interface of active and standby clusters, choose **O&M** > **Log** > **Download**.

Step 9 In the **Service** in the required cluster drop-down list box, select **HBase**.

Step 10 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.149 ALM-19011 RegionServer Region Number Exceeds the Threshold

Description

The system checks the number of regions on each RegionServer in each HBase service instance every 30 seconds. The region number is displayed on the HBase service monitoring page and RegionServer role monitoring page. This alarm is generated when the number of regions on a RegionServer exceeds the threshold (default value: 2000) for 20 consecutive times. The threshold can be changed by choosing **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **HBase**. This alarm is cleared when the number of regions is less than or equal to the threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Auto Clear
19011	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The data read/write performance of HBase is affected when the number of regions on a RegionServer exceeds the threshold.

Possible Causes

- The RegionServer region distribution is unbalanced.
- The HBase cluster scale is too small.

Procedure

View alarm location information.

Step 1 On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Alarms**, select this alarm, and view the service instance and host name in **Location**.

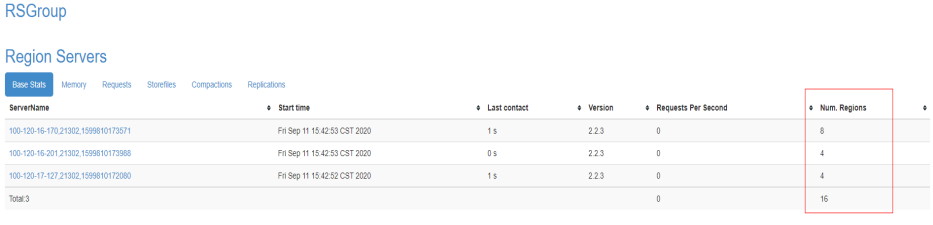
Step 2 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services**, click the HBase service instance for which the alarm is generated, and click **HMaster(Active)**. On the displayed WebUI of the HBase instance, check whether the region distribution on the RegionServer is balanced.

NOTE

By default, the **admin** user does not have the permissions to manage other components. If the page cannot be opened or the displayed content is incomplete when you access the native UI of a component due to insufficient permissions, you can manually create a user with the permissions to manage that component.

- If yes, go to **Step 9**.
- If no, go to **Step 3**.

Figure 10-38 WebUI of HBase instance



Enable load balancing.

Step 3 Log in to the node where the HBase client is located as user **root**. Go to the client installation directory, and set environment variables.

cd *client installation directory*

source bigdata_env

If the cluster adopts the security mode, perform security authentication. Specifically, run the **kinit hbase** command and enter the password as prompted (obtain the password from the administrator).

Step 4 Run the following commands to go to the HBase shell command window and check whether the load balancing function is enabled.

hbase shell

balancer_enabled

- If yes, go to **Step 6**.
- If no, go to **Step 5**.

Step 5 On the HBase shell command window, run the following commands to enable the load balancing function and check whether the function is enabled.

balance_switch true

balancer_enabled

Step 6 On the HBase shell command window, run the **balancer** command to manually trigger the load balancing function.

NOTE

You are advised to enable and manually trigger the load balancing function during off-peak hours.

Step 7 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**, and click **HMaster(Active)**. On the displayed WebUI of the HBase instance, refresh the page and check whether the region distribution is balanced.

- If yes, go to **Step 8**.
- If no, go to **Step 21**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Delete unwanted HBase tables.

NOTE

Exercise caution when deleting data to ensure data is deleted correctly.

Step 9 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**, and click **HMaster(Active)**. On the displayed WebUI of the HBase instance, view tables stored in the HBase service instance and record unwanted tables that can be deleted.

Step 10 On the HBase shell command window, run the **disable** command and **drop** command to delete the table to decrease the number of regions.

disable '*name of the table to be deleted***'**

drop '*name of the table to be deleted***'**

Step 11 On the HBase shell command window, run the following command to check whether the load balancing function is enabled.

balancer_enabled

- If yes, go to **Step 13**.
- If no, go to **Step 12**.

Step 12 On the HBase shell command window, run the following commands to enable the load balancing function and confirm that the function is enabled.

balance_switch true

balancer_enabled

Step 13 On the HBase shell command window, run the **balancer** command to manually trigger the load balancing function.

Step 14 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**, and click **HMaster(Active)**. On the displayed WebUI of the HBase instance, refresh the page and check whether the region distribution is balanced.

- If yes, go to **Step 15**.
- If no, go to **Step 21**.

Step 15 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 16**.

Adjust the threshold.

Step 16 On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **HBase** > **Regions(RegionServer)**, select the applied rule, and click **Modify** to check whether the threshold is proper.

- If it is excessively small, increase the threshold as required and go to **Step 17**.
- If it is proper, go to **Step 18**.

Step 17 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 18**.

Perform system capacity expansion.

Step 18 Add nodes to the HBase cluster and add RegionServer instances to the nodes. Then enable and manually trigger the load balancing function.

Step 19 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services**, click the HBase service instance for which the alarm is generated, and click **HMaster(Active)**. On the displayed WebUI of the HBase instance, refresh the page and check whether the region distribution is balanced.

- If yes, go to **Step 20**.
- If no, go to **Step 21**.

Step 20 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 21**.

Collect fault information.

Step 21 On the FusionInsight Manager home page of the active and standby clusters, choose **O&M**> **Log** > **Download**.

Step 22 Select **HBase** in the required cluster from the **Service**.

Step 23 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 24 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.150 ALM-19012 HBase System Table Directory or File Lost

Description

The system checks whether HBase directories and files exist on the HDFS every 120 seconds. This alarm is generated when the system detects that the files or directories do not exist. This alarm is cleared when the files or directories are restored.

The HBase directories and files are as follows:

- Directory of the namespace **hbase** on the HDFS
- **hbase.version** file
- Directory of the table **hbase:meta** on the HDFS, .tableinfo file, and .regioninfo file
- Directory of the table **hbase:namespace** on the HDFS, .tableinfo file, and .regioninfo file
- Directory of the table **hbase:hindex** on the HDFS, .tableinfo file, and .regioninfo file
- Directory of the **hbase:acl** table on the HDFS, .tableinfo, and .regioninfo file (This table does not exist in the common mode cluster by default.)

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19012	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The HBase service fails to restart or start.

Possible Causes

Files or directories on the HDFS are missing.

Procedure

Locate the alarm cause.

Step 1 On the FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Click this alarm and check whether **Alarm Cause** indicates unknown errors.

- If yes, go to **Step 4**.
- If no, go to **Step 2**

Step 2 On the FusionInsight Manager home page, choose **O&M** > **Backup and Restoration** > **Backup Management**. Check whether there are success records of the backup task named **default** or other HBase metadata backup tasks that have been successfully executed.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Use the latest backup metadata to restore the metadata of the HBase service.

Collect fault information.

Step 4 On the FusionInsight Manager page of the active and standby clusters, choose **O&M** > **Log** > **Download**.

Step 5 In the **Service** area, select faulty HBase services in the required cluster.

Step 6 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.151 ALM-19013 Duration of Regions in transaction State Exceeds the Threshold

Description

The system checks the number of regions in transaction state on HBase every 300 seconds. This alarm is generated when the system detects that the duration of regions in transaction state exceeds the threshold for two consecutive times. This alarm is cleared when all timeout regions are restored.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase service instances are installed, you need to determine the HBase service instance where the alarm is generated based on the value of **ServiceName** in **Location**. For example, if the HBase1 service is unavailable, **ServiceName=HBase1** is displayed in **Location**, and the operation object in the procedure needs to be changed from HBase to HBase1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19013	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Some data in the table gets lost or becomes unavailable.

Possible Causes

- Compaction is permanently blocked.
- The HDFS files are abnormal.

Procedure

Locate the alarm cause.

Step 1 On the FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**, select this alarm, and view the **HostName** and **RoleName** in **Location**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services > HBase**, Click the drop-down menu in the chartarea and choose **Customize > Service** >

Region in transaction count to view **Region in transaction count over threshold**. Check whether the monitoring item detects a value in three consecutive detection periods. (The default threshold is 60 seconds.)

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **HMaster (Active)** > **Tables** to check whether the regions of only one table transaction status time out.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 Run the **hbase hbck** command on the client and check whether the error message "No table descriptor file under hdfs://hacluster/hbase/data/default/table" is displayed.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 Log in to the client as user **root**. Run the following command:

cd *client installation directory*

source bigdata_env

If the cluster is in security mode, run the **kinit hbase** command

Log in to the HMaster WebUI, choose **Procedure & Locks** in the navigation tree, and check whether any process ID is in the **Waiting** state in **Procedures**. If yes, run the following command to release the procedure lock:

hbase hbck -j *client installation directory*/**HBase/hbase/tools/hbase-hbck2-*.jar bypass -o** *pid*

Check whether the state is in the **Bypass** state. If the procedure on the UI is always in **RUNNABLE(Bypass)** state, perform an active/standby switchover. Run the **assigns** command to bring the region online again.

hbase hbck -j *client installation directory*/**HBase/hbase/tools/hbase-hbck2-*.jar assigns -o** *regionName*

Step 6 Repeat **Step 4**. Run the **hbase hbck** command on the client and check whether the error message "No table descriptor file under hdfs://hacluster/hbase/data/default/table" is displayed.

- If yes, go to **Step 7**.
- If no, no further action is required.

Collect fault information.

Step 7 On the FusionInsight Manager page of the active and standby clusters, choose **O&M** > **Log** > **Download**.

Step 8 In the **Service** area, select faulty HBase services in the required cluster.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.152 ALM-19014 Capacity Quota Usage on ZooKeeper Exceeds the Threshold Severely

Description

The system checks the ZNode usage of HBase every 120 seconds. This alarm is generated when the ZNode capacity usage of HBase exceeds the "Critical" alarm threshold (90% by default).

This alarm is cleared when the capacity usage of ZNode is less than the "Critical" alarm threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase services are installed, determine the HBase service for which the alarm is generated based on the value of **ServiceName** in **Location** of the alarm. For example, if **ServiceName** is **HBase-1** in **Location**, the operation object in the handling procedure should be changed from HBase to HBase-1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19014	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Threshold	Specifies the threshold for which the alarm is generated.

Impact on the System

This alarm indicates that the capacity usage of the ZNode of HBase has exceeded the threshold severely. As a result, the write request of the HBase service fails.

Possible Causes

- DR is configured for HBase, and data synchronization fails or is slow in DR.
- A large number of WAL files are being split in the HBase cluster.

Procedure

Check the capacity configuration and usage of ZNode.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Select the alarm whose **Alarm ID** is "19014", and view the threshold in **Additional Information**.

Step 2 Log in to the HBase client as user **root**. Run the following command to access the client installation directory:

cd *client installation directory*

Run the following command to set environment variables:

source bigdata_env

If the cluster uses the security version, run the following command to perform security authentication:

kinit hbase

Enter the password as prompted (obtain the password from the administrator).

Step 3 Run the **hbase zkcli** command to log in to the ZooKeeper client and run the **listquota /hbase** command to view the ZNode capacity quota of the HBase service. The ZNode root directory in the command is specified by the **zookeeper.znode.parent** parameter of the HBase service. The marked area in the following figure shows the capacity configuration of the root ZNode of the HBase service.

```
[zk: 189-185-229-159:24002,189-185-229-114:24002,189-185-229-251:24002(CONNECTED) 145] listquota /hbase
absolute path is /zookeeper/quota/hbase
Output quota for /hbase count=1500000,bytes=10240
Output stat for /hbase count=42,bytes=1601
```

Step 4 Run the **getusage /hbase/splitWAL** command to check the capacity usage of the ZNode. Check whether the ratio of "Data size" in the command output to the ZNode capacity quota is close to the alarm threshold.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose **Alarm ID** is "12007", "19000", or "19013" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**.
- If no, go to **Step 9**

Step 6 Run the **getusage /hbase/replication** command to check the capacity usage of the ZNode. Check whether the ratio of "Data size" in the command output to the ZNode capacity quota is close to the alarm threshold.

- If yes, go to **Step 7**.

- If no, go to **Step 9**.

Step 7 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose **Alarm ID** is "19006" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**
- If no, go to **Step 9**.

Step 8 Wait for five minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager page, choose **O&M** > **Log** > **Download**.

Step 10 In **Service**, select **HBase** of the cluster to be operated.

Step 11 Click  in the upper right corner to set **Start Date** and **End Date** to 10 minutes before and after the time when the alarm is generated, and click **Download**.

Step 12 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.153 ALM-19015 Quantity Quota Usage on ZooKeeper Exceeds the Threshold

Description

The system checks the ZNode usage of the HBase service every 120 seconds. This alarm is generated when the system detects that the quantity of ZNodes used of the HBase service exceeds the alarm threshold (75% by default).

This alarm is cleared when the quantity usage of ZNodes is less than the threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase services are installed, determine the HBase service for which the alarm is generated based on the value of **ServiceName** in **Location** of the alarm. For example, if **Service name** is **HBase-1** in **Location**, change the operation object in the handling procedure from HBase to HBase-1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19015	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Threshold	Specifies the threshold for which the alarm is generated.

Impact on the System

This alarm indicates that the quantity of ZNodes used in the HBase service has exceeded the threshold. If this alarm is not handled in a timely manner, the problem severity may be escalated to "Critical", affecting data writing.

Possible Causes

- DR is configured for HBase, and data synchronization fails or is slow in DR.
- A large number of WAL files are being split in the HBase cluster.

Procedure

Check the quantity quota and usage of ZNodes.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Select the alarm whose ID is "19015", and view the threshold in **Additional Information**.

Step 2 Log in to the HBase client as user **root**. Run the following command to access the client installation directory:

cd *client installation directory*

Run the following command to set environment variables:

source bigdata_env

If the cluster uses the security version, run the following command to perform security authentication:

kinit hbase

Enter the password as prompted (obtain the password from the administrator).

Step 3 Run the **hbase zkcli** command to log in to the ZooKeeper client and run the **listquota /hbase** command to view the ZNode quantity quota of the HBase service. The ZNode root directory in the command is specified by the **zookeeper.znode.parent** parameter of the HBase service. The marked area in the following figure shows the quantity quota of the root ZNode of the HBase service.

```
[zk: 189-185-229-159:24002,189-185-229-114:24002,189-185-229-251:24002(CONNECTED) 7] listquota /hbase
absolute path is /zookeeper/quota/hbase
Output quota for /hbase count=1500000,bytes=10240
Output stat for /hbase count=59,bytes=1902
```

Step 4 Run the **getusage /hbase/splitWAL** command to check the quantity usage of the ZNode. Check whether the ratio of "Node count" in the command output to the ZNode quantity quota is close to the alarm threshold.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose ID is "12007", "19000", or "19013" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**.
- If no, go to **Step 9**.

Step 6 Run the **getusage /hbase/replication** command to check the usage of the ZNodes. Check whether the ratio of "Node count" in the command output to the ZNode quantity quota is close to the alarm threshold.

- If yes, go to **Step 7**.
- If no, go to **Step 9**.

Step 7 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose Alarm ID is "19006" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 7**.
- If no, go to **Step 9**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 In the **Service** area, select **HBase** of the cluster to be operated.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.154 ALM-19016 Quantity Quota Usage on ZooKeeper Exceeds the Threshold Severely

Alarm description

The system checks the ZNode usage of HBase every 120 seconds. This alarm is generated when the ZNode quantity usage of HBase exceeds the "Critical" alarm threshold (90% by default).

This alarm is cleared when the quantity usage of the ZNode is less than the "Critical" alarm threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase services are installed, determine the HBase service for which the alarm is generated based on the value of **ServiceName** in **Location** of the alarm. For example, if **Service name** is **HBase-1** in **Location**, change the operation object in the handling procedure from HBase to HBase-1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19016	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Threshold	Specifies the threshold for which the alarm is generated.

Impact on the System

This alarm indicates that the quantity usage of the ZNode of HBase has exceeded the threshold severely. As a result, the write request of the HBase service fails.

Possible Causes

- DR is configured for HBase, and data synchronization fails or is slow in DR.
- A large number of WAL files are being split in the HBase cluster.

Procedure

Check the quantity quota and usage of ZNodes.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Select the alarm whose ID is "19016", and view the threshold in **Additional Information**.

Step 2 Log in to the HBase client as user **root**. Run the following command to access the client installation directory:

cd *client installation directory*

Run the following command to set environment variables:

source bigdata_env

If the cluster uses the security version, run the following command to perform security authentication:

kinit hbase

Enter the password as prompted (obtain the password from the administrator).

Step 3 Run the **hbase zkcli** command to log in to the ZooKeeper client and run the **listquota /hbase** command to view the ZNode quantity quota of the HBase service. The ZNode root directory in the command is specified by the **zookeeper.znode.parent** parameter of the HBase service. The marked area in the following figure shows the quantity configuration of the root ZNode of the HBase service.

```
[zk: 189-185-229-159:24002,189-185-229-114:24002,189-185-229-251:24002(CONNECTED) 7] listquota /hbase
absolute path is /zookeeper/quota/hbase
Output quota for /hbase count=1500000,bytes=10240
Output stat for /hbase count=59,bytes=1902
```

Step 4 Run the **getusage /hbase/splitWAL** command to check the quantity usage of the ZNode. Check whether the ratio of "Node count" in the command output to the ZNode quantity quota is close to the alarm threshold.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose ID is "12007", "19000", or "19013" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**.
- If no, go to **Step 9**.

Step 6 Run the **getusage /hbase/replication** command to check the usage of the ZNodes. Check whether the ratio of "Node count" in the command output to the ZNode quantity quota is close to the alarm threshold.

- If yes, go to **Step 7**.
- If no, go to **Step 9**.

Step 7 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose Alarm ID is "19006" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 In the **Service** area, select **HBase** of the cluster to be operated.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.155 ALM-19017 Capacity Quota Usage on ZooKeeper Exceeds the Threshold

Description

The system checks the ZNode usage of the HBase service every 120 seconds. This alarm is generated when the system detects that the ZNodes capacity usage of the HBase service exceeds the alarm threshold (75% by default).

This alarm is cleared when the capacity usage of the ZNode capacity is less than the threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase services are installed, determine the HBase service for which the alarm is generated based on the value of **ServiceName** in **Location** of the alarm. For example, if **Service name** is **HBase-1** in **Location**, change the operation object in the handling procedure from HBase to HBase-1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19017	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Threshold	Specifies the threshold for which the alarm is generated.

Impact on the System

This alarm indicates that the ZNodes capacity usage in the HBase service has exceeded the threshold. If this alarm is not handled in a timely manner, the problem severity may be escalated to "Critical", affecting data writing.

Possible Causes

- DR is configured for HBase, and data synchronization fails or is slow in DR.
- A large number of WAL files are being split in the HBase cluster.

Procedure

Check the capacity configuration and usage of ZNode.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Select the alarm whose ID is "19017", and view the threshold in **Additional Information**.

Step 2 Log in to the HBase client as user **root**. Run the following command to access the client installation directory:

cd *client installation directory*

Run the following command to set environment variables:

source bigdata_env

If the cluster uses the security version, run the following command to perform security authentication:

kinit hbase

Enter the password as prompted (obtain the password from the administrator).

Step 3 Run the **hbase zkcli** command to log in to the ZooKeeper client and run the **listquota /hbase** command to view the ZNode capacity quota of the HBase service. The ZNode root directory in the command is specified by the **zookeeper.znode.parent** parameter of the HBase service. The marked area in the following figure shows the capacity configuration of the root ZNode of the HBase service.

```
[zk: 189-185-229-159:24002,189-185-229-114:24002,189-185-229-251:24002(CONNECTED) 145] listquota /hbase
absolute path is /zookeeper/quota/hbase
Output quota for /hbase count=1500000,bytes=10240
Output stat for /hbase count=42,bytes=1601
```

Step 4 Run the **getusage /hbase/splitWAL** command to check the capacity usage of the ZNode. Check whether the ratio of "Data size" to the ZNode capacity quota is close to the alarm threshold.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 On the FusionInsight Manager, check whether the alarm whose ID is "12007", "19000", or "19013" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**.
- If no, go to **Step 7**.

Step 6 Run the **getusage /hbase/replication** command to check the capacity usage of the ZNode. Check whether the ratio of "Data size" to the ZNode capacity quota is close to the alarm threshold.

- If yes, go to **Step 7**.

- If no, go to **Step 9**.

Step 7 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether the alarm whose Alarm ID is "19006" and **ServiceName** in **Location** is the current HBase service exists.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 In the **Service** area, select **HBase** of the cluster to be operated.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.156 ALM-19018 HBase Compaction Queue Exceeds the Threshold

Description

The system checks the compaction queue length of the HBase service every 300 seconds. This alarm is generated when the system detects that the compaction queue length of the HBase service exceeds the alarm threshold (100 by default). This alarm is cleared when the compaction queue length is less than the alarm threshold.

NOTE

If the multi-instance function is enabled in the cluster and multiple HBase services are installed, determine the HBase service for which the alarm is generated based on the value of **ServiceName** in **Location** of the alarm. For example, if **Service name** is **HBase-1** in **Location**, change the operation object in the handling procedure from HBase to HBase-1.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
19018	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The compaction queue length of the HBase service exceeds the threshold. If this alarm is not handled in a timely manner, the cluster performance may deteriorate, affecting data read and write.

Possible Causes

- The number of HBase RegionServers is too small.
- There are too many regions on a single RegionServer of HBase.
- The HBase RegionServer heap size is small.
- Resources are insufficient.
- The related parameters are set improperly.

Procedure

Check whether the related configuration is correct.

Step 1 On the FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** and check whether alarms "19011" exist.

- If yes, click **View Help** on the right of the alarm and rectify the fault by referring to the help document. Then, go to **Step 3**.
- If no, go to **Step 2**.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations** > **All Configurations**. Search for **hbase.hstore.compaction.min**, **hbase.hstore.compaction.max**, **hbase.hstore.compactionThreshold**,

hbase.regionserver.thread.compaction.small, and **hbase.regionserver.thread.compaction.throttle**, and increase their values.

Step 3 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 In the **Service** area, select **HBase** of the cluster to be operated.

Step 6 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.157 ALM-19019 Number of HBase HFiles to Be Synchronized Exceeds the Threshold

Description

The system checks the number of HFiles to be synchronized by the RegionServer of each HBase service instance every 30 seconds. This indicator can be viewed on the RegionServer role monitoring page. This alarm is generated when the number of HFiles to be synchronized on a RegionServer exceeds the threshold (exceeding 128 for 20 consecutive times by default). To change the threshold, choose **O&M** > **Alarm** > **Threshold Configuration** > *Name of the desired cluster* > **HBase** . This alarm is cleared when the number of HFiles to be synchronized is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
19019	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the number of HFiles to be synchronized by a RegionServer exceeds the threshold, the number of ZNodes used by HBase exceeds the threshold, affecting the HBase service status.

Possible Causes

- The network is abnormal.
- The RegionServer region distribution is unbalanced.
- The HBase service scale of the standby cluster is too small.

Procedure

View alarm location information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing the alarm whose **Alarm ID** is **19019**, and view the service instance and host name in **Location**.

Check the network connection between RegionServers on active and standby clusters.

Step 2 Run the **ping** command to check whether the network connection between the faulty RegionServer node and the host where RegionServer of the standby cluster resides is normal.

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 Contact the network administrator to restore the network.

Step 4 After the network recovers, check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 5**.

Check the RegionServer region distribution in the active cluster.

Step 5 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**. Click **HMaster(Active)** to go to the web UI of the HBase instance and check whether regions are evenly distributed on the Region Server.

Step 6 Log in to the faulty RegionServer node as user **omm**.

Step 7 Run the following commands to go to the client installation directory and set the environment variable:

cd *Client installation directory*

source bigdata_env

If the cluster uses the security mode, perform security authentication. Run the **kinit hbase** command and enter the password as prompted (obtain the password from the administrator).

Step 8 Run the following commands to check whether the load balancing function is enabled.

hbase shell

balancer_enabled

- If yes, go to **Step 10**.
- If no, go to **Step 9**.

Step 9 Run the following commands in HBase Shell to enable the load balancing function and check whether the function is enabled.

balance_switch true

balancer_enabled

Step 10 Run the **balancer** command to manually trigger the load balancing function.

NOTE

You are advised to enable and manually trigger the load balancing function during off-peak hours.

Step 11 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Check the HBase service scale of the standby cluster.

Step 12 Expand the HBase cluster, add a node, and add a RegionServer instance on the node. Then, perform **Step 6** to **Step 10** to enable the load balancing function and manually trigger it.

Step 13 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**. Click **HMaster(Active)** to go to the web UI of the HBase instance, refresh the page, and check whether regions are evenly distributed.

- If yes, go to **Step 14**.

- If no, go to **Step 15**.

Step 14 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Collect the fault information.

Step 15 On FusionInsight Manager of the standby cluster, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 16 Expand the **Service** drop-down list, and select **HBase** for the target cluster.

Step 17 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.158 ALM-19020 Number of HBase WAL Files to Be Synchronized Exceeds the Threshold

Description

The system checks the number of WAL files to be synchronized by the RegionServer of each HBase service instance every 30 seconds. This indicator can be viewed on the RegionServer role monitoring page. This alarm is generated when the number of WAL files to be synchronized on a RegionServer exceeds the threshold (exceeding 128 for 20 consecutive times by default). To change the threshold, choose **O&M** > **Alarm** > **Threshold Configuration** > *Name of the desired cluster* > **HBase** . This alarm is cleared when the number of WAL files to be synchronized is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
19020	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the number of WAL files to be synchronized by a RegionServer exceeds the threshold, the number of ZNodes used by HBase exceeds the threshold, affecting the HBase service status.

Possible Causes

- The network is abnormal.
- The RegionServer region distribution is unbalanced.
- The HBase service scale of the standby cluster is too small.

Procedure

View alarm location information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing the alarm whose **Alarm ID** is **19020**, and view the service instance and host name in **Location**.

Check the network connection between RegionServers on active and standby clusters.

Step 2 Run the **ping** command to check whether the network connection between the faulty RegionServer node and the host where RegionServer of the standby cluster resides is normal.

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 Contact the network administrator to restore the network.

Step 4 After the network recovers, check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 5**.

Check the RegionServer region distribution in the active cluster.

Step 5 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**. Click **HMaster(Active)** to go to the web UI of the HBase instance and check whether regions are evenly distributed on the Region Server.

Step 6 Log in to the faulty RegionServer node as user **omm**.

Step 7 Run the following commands to go to the client installation directory and set the environment variable:

cd *Client installation directory*

source bigdata_env

If the cluster uses the security mode, perform security authentication. Run the **kinit hbase** command and enter the password as prompted (obtain the password from the administrator).

Step 8 Run the following commands to check whether the load balancing function is enabled.

hbase shell

balancer_enabled

- If yes, go to **Step 10**.
- If no, go to **Step 9**.

Step 9 Run the following commands in HBase Shell to enable the load balancing function and check whether the function is enabled.

balance_switch true

balancer_enabled

Step 10 Run the **balancer** command to manually trigger the load balancing function.

NOTE

You are advised to enable and manually trigger the load balancing function during off-peak hours.

Step 11 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 12**.

Check the HBase service scale of the standby cluster.

Step 12 Expand the HBase cluster, add a node, and add a RegionServer instance on the node. Then, perform **Step 6** to **Step 10** to enable the load balancing function and manually trigger it.

Step 13 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase**. Click **HMaster(Active)** to go to the web UI of the HBase instance, refresh the page, and check whether regions are evenly distributed.

- If yes, go to **Step 14**.

- If no, go to **Step 15**.

Step 14 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Collect the fault information.

Step 15 On FusionInsight Manager of the standby cluster, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 16 Expand the **Service** drop-down list, and select **HBase** for the target cluster.

Step 17 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.159 ALM-20002 Hue Service Unavailable

Description

This alarm is generated when the Hue service is unavailable. The system checks the Hue service status every 60 seconds.

This alarm is cleared when the Hue service is normal.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
20002	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The system cannot provide data loading, query, and extraction services.

Possible Causes

- The internal KrbServer service on which the Hue service depends is abnormal.
- The internal DBService service on which the Hue service depends is abnormal.
- The network connection to the DBService is abnormal.

Procedure

Check whether the KrbServer is abnormal.

Step 1 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services**. In the service list, check whether the **KrbServer** running status is **Normal**.

- If yes, go to **Step 4**.
- If no, go to **Step 2**.

Step 2 Restart the KrbServer service.

Step 3 Wait several minutes, and check whether **Hue Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the DBService is abnormal.

Step 4 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 5 In the service list, check whether the **DBService** running status is **Normal**.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 Restart the DBService.

NOTE

To restart the service, enter the FusionInsight Manager administrator password.

Step 7 Wait several minutes, and check whether **Hue Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check whether the network connection to the DBService is normal.

Step 8 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hue** > **Instance**, record the IP address of the active Hue.

Step 9 Log in to the active Hue.

Step 10 Run the **ping** command to check whether communication between the host that runs the active Hue and the hosts that run the DBService is normal. (Obtain the IP addresses of the hosts that run the DBService in the same way as that for obtaining the IP address of the active Hue.)

- If yes, go to **Step 13**.
- If no, go to **Step 11**.

Step 11 Contact the administrator to restore the network.

Step 12 Wait several minutes, and check whether **Hue Service Unavailable** is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collect fault information.

Step 13 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 14 Select the following nodes in the required cluster from the **Service** drop-down list:

- Hue
- Controller

Step 15 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 On the FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hue**.

Step 17 Choose **More** > **Restart Service**, and click **OK**.

Step 18 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 19**.

Step 19 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.160 ALM-24000 Flume Service Unavailable

Description

The alarm module checks the Flume service status every 180 seconds. This alarm is generated if the Flume service is abnormal.

This alarm is automatically cleared after the Flume service recovers.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Flume cannot work and data transmission is interrupted.

Possible Causes

All Flume instances are faulty.

Procedure

Step 1 Log in to a Flume node as user **omm** and run the **ps -ef|grep "flume.role=server"** command to check whether the Flume process exists on the node.

- If yes, go to **Step 3**.

- If no, restart the faulty Flume node or Flume service and go to **Step 2**.

Step 2 In the alarm list, check whether alarm "Flume Service Unavailable" is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Collect the fault information.

Step 3 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 4 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 5 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 6 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.161 ALM-24001 Flume Agent Exception

Description

The Flume agent instance for which the alarm is generated cannot be started. This alarm is generated when the Flume agent process is faulty (The system checks in every 5 seconds.) or Flume agent fails to start (The system reporting alarms immediately).

This alarm is cleared when the Flume agent process recovers, Flume agent starts successfully and the alarm handling is completed.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24001	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
AgentId	Specifies the ID of the agent for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Flume agent instance for which the alarm is generated cannot provide services properly, and the data transmission tasks of the instance are temporarily interrupted. Real-time data is lost during real-time data transmission.

Possible Causes

- The JAVA_HOME directory does not exist or the Java permission is incorrect.
- The Flume agent directory permission is incorrect.
- Flume agent fails to start.

Procedure

Check whether the JAVA_HOME directory exists or whether the JAVA permission is correct.

Step 1 Log in to the host for which the alarm is generated as user **root**.

Step 2 Run the following command to obtain the installation directory of the Flume client for which the alarm is generated: (The value of **AgentId** can be obtained from **Location** of the alarm.)

ps -ef|grep *AgentId* **| grep -v grep | awk -F 'conf-file ' '{print $2}' | awk -F 'fusioninsight' '{print $1}'**

Step 3 Run the **su -** *Flume installation user* command to switch to the Flume installation user and run the **cd** *Flume client installation directory*/**fusioninsight-flume-1.9.0/conf/** command to go to the Flume configuration directory.

Step 4 Run the **cat ENV_VARS | grep JAVA_HOME** command.

Step 5 Check whether the **JAVA_HOME** directory exists. If the command output in **Step 4** is not empty and **ll $JAVA_HOME/** is not empty, the **JAVA_HOME** directory exists.

- If yes, go to **Step 7**.

- If no, go to **Step 6**.

Step 6 Specify a correct **JAVA_HOME** directory.

Step 7 Run the **$JAVA_HOME/bin/java -version** command to check whether the Flume agent running user has the Java execution permission. If the Java version is displayed in the command output, the Java permission meets the requirement. Otherwise, the Java permission does not meet the requirement.

- If yes, go to **Step 9**.
- If no, go to **Step 8**.

NOTE

JAVA_HOME is the environment variable exported during Flume client installation. You can also go to *Flume client installation directory*/**fusioninsight-flume-1.9.0/conf** and run the **cat ENV_VARS | grep JAVA_HOME** command to view the variable value.

Step 8 Run the **chmod 750 $JAVA_HOME/bin/java** command to grant the Java execution permission to the Flume agent running user.

Check the directory permission of the Flume agent.

Step 9 Log in to the host for which the alarm is generated as user **root**.

Step 10 Run the following command to switch to the Flume agent installation directory:

cd *Flume client installation directory*/**fusioninsight-flume-1.9.0/conf/**

Step 11 Run the **ls -al * -R** command to check whether any file owner is the user who running the Flume agent.

- If yes, go to **Step 12**.
- If no, run the **chown** command to change the file owner to the user who runs the Flume agent.

Check the Flume agent configuration.

Step 12 Run the **cat properties.properties | grep spooldir** and **cat properties.properties | grep TAILDIR** commands to check whether the Flume source type is spoolDir or tailDir. If any command output is displayed, the Flume source type is spoolDir or tailDir.

- If yes, go to **Step 13**.
- If no, go to **Step 17**.

Step 13 Check whether the data monitoring directory exists.

- If yes, go to **Step 15**.
- If no, go to **Step 14**.

NOTE

Run the **cat properties.properties | grep spoolDir** command to view the spoolDir monitoring directory.

```
[root@[redacted] /tmp/test/fusioninsight-flume-1.9.0/conf]# cat properties.properties | grep spoolDir
client.sources.aa1.spoolDir = /opt/liuxingcheng/flumeclient/sourcedata/flumesourcedata1
[root@[redacted] /tmp/test/fusioninsight-flume-1.9.0/conf]#
```

Run the **cat properties.properties | grep parentDir** command to view the tailDir monitoring directory.

```
[root@[redacted] /tmp/test/fusioninsight-flume-1.9.0/conf]# cat properties.properties | grep parentDir
server.sources.AAAA.filegroups.F1.parentDir = /tmp/flumetest/taildir_data
[root@[redacted] /tmp/test/fusioninsight-flume-1.9.0/conf]#
```

Step 14 Specify a correct data monitoring directory.

Step 15 Check whether the Flume agent user has the read, write, and execute permissions on the monitoring directory specified in **Step 13**.

- If yes, go to **Step 17**.
- If no, go to **Step 16**.

NOTE

Go to the monitoring directory as the Flume running user. If files can be created, the Flume running user has the read, write, and execute permissions on the monitoring directory.

Step 16 Run the **chmod 777** *Flume monitoring directory* command to grant the Flume agent running user the read, write, and execute permissions on the monitoring directory specified in **Step 13**.

Step 17 Check whether the components connected to the Flume sink are in safe mode.

- If yes, go to **Step 18**.
- If no, go to **Step 23**.

NOTE

If the sinks in the **properties.properties** configuration file are the HDFS sink and HBase sink, and the configuration file contains a keytab file, the components connected to the Flume sink are in safe mode.

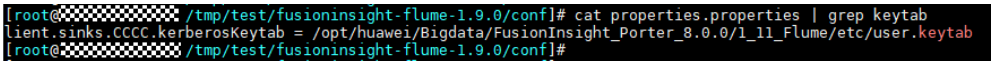
If the sink in the **properties.properties** configuration file is the kafka sink and ***.security.protocol** is set to **SASL_PLAINTEXT** or **SASL_SSL**, Kafka connected to the Flume sink is in safe mode.

Step 18 Run the **ll** *ketab path* command to check whether the keytab authentication path specified by the ***.kerberosKeytab** parameter in the configuration file exists.

- If yes, go to **Step 20**.
- If no, go to **Step 19**.

NOTE

To view the ketab path, run the **cat properties.properties | grep keytab** command.



Step 19 Change the value of **kerberosKeytab** in **Step 18** to the custom keytab path and go to **Step 21**.

Step 20 Perform **Step 18** to check whether the Flume agent running user has the permission to access the keytab authentication file. If the keytab path is returned, the user has the permission. Otherwise, the user does not have the permission.

- If yes, go to **Step 22**.
- If no, go to **Step 21**.

Step 21 Run the **chmod 755** *ketab file* command to grant the read permission on the keytab file specified in **Step 19**, and restart the Flume process.

Step 22 Check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 23**.

Collect the fault information.

Step 23 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 24 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 25 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 26 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.162 ALM-24003 Flume Client Connection Interrupted

Description

The alarm module monitors the port connection status on the Flume server. This alarm is generated if the Flume server fails to receive a connection message from the Flume client in three consecutive minutes.

This alarm is cleared after the Flume server receives a connection message from the Flume client.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24003	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
Client IP Address	Specifies the IP address of the Flume client.

Name	Meaning
Client Name	Specifies the agent name of the Flume client.
Sink Name	Specifies the sink name of Flume Agent.

Impact on the System

The communication between the Flume client and the server fails. The Flume client cannot send data to the Flume server.

Possible Causes

- The network connection between the Flume client and the server is faulty.
- The Flume client's process is abnormal.
- The Flume client is incorrectly configured.

Procedure

Check the network connection between the Flume client and the server.

Step 1 Log in to the host whose IP address is specified by **Flume ClientIP** in the alarm information as user **root**.

Step 2 Run the **ping** *Flume server IP address* command to check whether the network connection between the Flume client and the server is normal.

- If yes, go to **Step 3**.
- If no, go to **Step 11**.

Check whether the Flume client's process is normal.

Step 3 Log in to the host whose IP address is specified by **Flume ClientIP** in the alarm information as user **root**.

Step 4 Run the **ps -ef|grep flume |grep client** command to check whether the Flume client process exists.

- If yes, go to **Step 5**.
- If no, go to **Step 11**.

Check the Flume client configuration.

Step 5 Log in to the host whose IP address is specified by **Flume ClientIP** in the alarm information as user **root**.

Step 6 Run the **cd** *Flume client installation directory*/**fusioninsight-flume-1.9.0/conf/** command to go to Flume's configuration directory.

Step 7 Run the **cat properties.properties** command to query the current configuration file of the Flume client.

Step 8 Check whether the **properties.properties** file is correctly configured according to the configuration description of the Flume agent.

- If yes, go to **Step 9**.
- If no, go to **Step 11**.

Step 9 Modify the **properties.properties** configuration file.

Check whether the alarm is cleared.

Step 10 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect the fault information.

Step 11 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 12 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 13 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Collect logs in the **/var/log/Bigdata/flume-client** directory on the Flume client using a transmission tool.

Step 15 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.163 ALM-24004 Exception Occurs When Flume Reads Data

Description

The alarm module monitors the status of Flume Source. This alarm is generated immediately when the duration in which Source fails to read the data exceeds the threshold.

The default threshold is **0**, indicating that the threshold is disabled. You can change the threshold by modifying the **properties.properties** file in the **conf** directory. Specifically, modify the **NoDatatime** parameter of required the source.

The alarm is cleared when Source reads the data and the alarm handling is complete.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24004	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
AgentId	Specifies the ID of the agent for which the alarm is generated.
ComponentType	Specifies the component type for which the alarm is generated.
ComponentName	Specifies the component name for which the alarm is generated.

Impact on the System

If data is found in the data source and Flume Source continuously fails to read data, the data collection is stopped.

Possible Causes

- Flume Source is faulty, so data cannot be sent.
- The network is faulty, so the data cannot be sent.

Procedure

Check whether Flume Source is faulty.

Step 1 Open the **properties.properties** configuration file on the local PC, search for **keyword type = spooldir** in the file, and check whether the Flume source type is spoolDir.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 View the spoolDir directory to check whether all files are already transferred.

- If yes, no further action is required.

- If no, go to **Step 5**.

  NOTE

  The monitoring directory of spooDir is specified by the **.spoolDir** parameter in the **properties.properties** configuration file. If all files in the monitoring directory have been transferred, the file name extension of all files in the monitoring directory is **.COMPLETED**.

Step 3 Open the **properties.properties** configuration file on the local PC, search for **org.apache.flume.source.kafka.KafkaSource** in the file, and check whether the Flume source type is Kafka.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 Check whether the topic data configured by Kafka Source has been used up.

- If yes, no further action is required.
- If no, go to **Step 5**.

Step 5 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance**.

Step 6 Go to the Flume instance page of the faulty node to check whether the indicator **Source Speed Metrics** in the alarm is 0.

- If yes, go to **Step 11**.
- If no, go to **Step 7**.

Check the network connection between the faulty node and the node that corresponds to the Flume Source IP address.

Step 7 Open the **properties.properties** configuration file on the local PC, search for **type = avro** in the file, and check whether the Flume source type is Avro.

- If yes, go to **Step 8**.
- If no, go to **Step 11**.

Step 8 Log in to the faulty node as user **root**, and run the **ping** *IP address of the Flume source* command to check whether the peer host can be pinged successfully.

- If yes, go to **Step 11**.
- If no, go to **Step 9**.

Step 9 Contact the network administrator to restore the network.

Step 10 In the alarm list, check whether the alarm is cleared after a period.

- If yes, no further action is required.
- If no, go to **Step 11**.

Collect the fault information.

Step 11 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 12 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 13 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 14 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.164 ALM-24005 Exception Occurs When Flume Transmits Data

Description

The alarm module monitors the capacity status of Flume Channel. The alarm is generated immediately when the duration that Channel is fully occupied exceeds the threshold or the number of times that Source fails to send data to Channel exceeds the threshold.

The default threshold is **10**. You can change the threshold by modifying the **channelfullcount** parameter of the related channel in the **properties.properties** configuration file in the **conf** directory.

The alarm is cleared when the space of Flume Channel is released and the alarm handling is complete.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
AgentId	Specifies the ID of the agent for which the alarm is generated.
ComponentType	Specifies the type of the component for which the alarm is generated.
ComponentName	Specifies the component for which the alarm is generated.

Impact on the System

If the disk usage of Flume Channel increases continuously, the time required for importing data to a specified destination prolongs. When the disk usage of Flume Channel reaches 100%, the Flume agent process pauses.

Possible Causes

- Flume Sink is faulty, so the data cannot be sent.
- The network is faulty, so the data cannot be sent.

Procedure

Check whether Flume Sink is faulty.

Step 1 Open the **properties.properties** configuration file on the local PC, search for **type = hdfs** in the file, and check whether the Flume sink type is HDFS.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 On FusionInsight Manager, check whether **HDFS Service Unavailable** alarm is generated in the alarm list and whether the HDFS service is stopped in the service list.

- If the alarm is reported, clear it according to the handling suggestions of ALM-14000 HDFS Service Unavailable; if the HDFS service is stopped, start it. Then go to **Step 7**.
- If no, go to **Step 7**.

Step 3 Open the **properties.properties** configuration file on the local PC, search for **type = hbase** in the file, and check whether the Flume sink type is HBase.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 On FusionInsight Manager, check whether **HBase Service Unavailable** alarm is generated in the alarm list and whether the HBase service is stopped in the service list.

- If the alarm is reported, clear it according to the handling suggestions of ALM-19000 HBase Service Unavailable; if the HBase service is stopped, start it. Then go to **Step 7**.

- If no, go to **Step 7**.

Step 5 Open the **properties.properties** configuration file on the local PC, search for **org.apache.flume.sink.kafka.KafkaSink** in the file, and check whether the Flume sink type is Kafka.

- If yes, go to **Step 6**.
- If no, go to **Step 9**.

Step 6 On FusionInsight Manager, check whether **Kafka Service Unavailable** alarm is generated in the alarm list and whether the Kafka service is stopped in the service list.

- If the alarm is reported, clear it according to the handling suggestions of ALM-38000 Kafka Service Unavailable; if the Kafka service is stopped, start it. Then go to **Step 7**.
- If no, go to **Step 7**.

Step 7 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance**.

Step 8 Go to the Flume instance page of the faulty node to check whether the indicator **Sink Speed Metrics** is 0.

- If yes, go to **Step 13**.
- If no, go to **Step 9**.

Check the network connection between the faulty node and the node that corresponds to the Flume Sink IP address.

Step 9 Open the **properties.properties** configuration file on the local PC, search for **type = avro** in the file, and check whether the Flume sink type is Avro.

- If yes, go to **Step 10**.
- If no, go to **Step 13**.

Step 10 Log in to the faulty node as user **root**, and run the **ping** *IP address of the Flume sink* command to check whether the peer host can be pinged successfully.

- If yes, go to **Step 13**.
- If no, go to **Step 11**.

Step 11 Contact the network administrator to restore the network.

Step 12 In the alarm list, check whether the alarm is cleared after a period.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collect the fault information.

Step 13 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 14 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 15 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.165 ALM-24006 Heap Memory Usage of Flume Server Exceeds the Threshold

Description

The system checks the heap memory usage of the Flume service every 60 seconds. This alarm is generated when the heap memory usage of the Flume instance exceeds the threshold (95% of the maximum memory) for 10 consecutive times. This alarm is cleared when the heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Heap memory overflow may cause service breakdown.

Possible Causes

The heap memory of the Flume instance is overused or the heap memory is inappropriately allocated.

Procedure

Check the heap memory usage.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **Flume Heap Memory Usage Exceeds the Threshold**, and view the **Location** information. Check the name of the host for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the target cluster* > **Services** > **Flume**. On the page that is displayed, click the **Instance** tab. On the displayed tab page, select the role corresponding to the host name for which the alarm is generated and select **Customize** from the drop-down list in the upper right corner of the chart area. Choose **Agent** and select **Flume Heap Memory Resource Percentage**. Then, click **OK**.

Step 3 Check whether the heap memory used by Flume reaches the threshold (95% of the maximum heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Flume** > **Configuration**. On the page that is displayed, click **All Configurations** and choose **Flume** > **System**. Set **-Xmx** in the **GC_OPTS** parameter to a larger value based on site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for the Flume server is insufficient for data transmission. You are advised to change the heap memory to: Channel capacity x Maximum size of a single data record x Number of channels. Note that the value of **xmx** cannot exceed the remaining memory of the node.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.166 ALM-24007 Flume Server Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the Flume service every 60 seconds. This alarm is generated when the direct memory usage of the Flume instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the Flume direct memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Direct memory overflow may cause service breakdown.

Possible Causes

The direct memory of the Flume process is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **Flume Direct Memory Usage Exceeds the Threshold**, and view the **Location** information. Check the name of the host for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the target cluster* > **Services** > **Flume**. On the page that is displayed, click the **Instance** tab. On the displayed tab page, select the role corresponding to the host name for which the alarm is generated and select **Customize** from the drop-down list in the upper right corner of the chart area. Choose **Agent** and select **Flume Direct Memory Resource Percentage**. Then, click **OK**.

Step 3 Check whether the direct memory used by Flume reaches the threshold (80% of the maximum direct memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Flume** > **Configuration**. On the page that is displayed, click **All Configurations** and choose **Flume** > **System**. Set **-XX:MaxDirectMemorySize** in the **GC_OPTS** parameter to a larger value based on site requirements and save the configuration.

NOTE

If this alarm is generated, the direct memory size configured for the Flume server instance cannot meet service requirements. You are advised to change the value of **-XX:MaxDirectMemorySize** to twice the current direct memory size or change the value based on site requirements.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.167 ALM-24008 Flume Server Non-Heap Memory Usage Exceeds the Threshold

Description

The system checks the non-heap memory usage of the Flume service every 60 seconds. This alarm is generated when the non-heap memory usage of the Flume instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the non-heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Non-heap memory overflow may cause service breakdown.

Possible Causes

The non-heap memory of the Flume instance is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **Flume Non-Heap Memory Usage Exceeds the Threshold**, and view the **Location** information. Check the name of the host for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the target cluster* > **Services** > **Flume**. On the page that is displayed, click the **Instance** tab. On the displayed tab page, select the role corresponding to the host name for which the alarm is generated and select **Customize** from the drop-down list in the upper right corner of the chart area. Choose **Agent** and select **Flume Non Heap Memory Resource Percentage**. Then, click **OK**.

Step 3 Check whether the non-heap memory used by Flume reaches the threshold (80% of the maximum non-heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Flume** > **Configuration**. On the page that is displayed, click **All Configurations** and choose **Flume** > **System**. Set **-XX: MaxPermSize** in the **GC_OPTS** parameter to a larger value based on site requirements and save the configuration.

NOTE

If this alarm is generated, the non-heap memory size configured for the Flume server instance cannot meet service requirements. You are advised to change the value of **-XX:MaxPermSize** to twice the current non-heap memory size or change the value based on site requirements.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.168 ALM-24009 Flume Server Garbage Collection (GC) Time Exceeds the Threshold

Description

The system checks the GC duration of the Flume process every 60 seconds. This alarm is generated when the GC duration of the Flume process exceeds the threshold (12 seconds by default) for five consecutive times. This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Flume data transmission efficiency decreases.

Possible Causes

The heap memory of the Flume process is overused or inappropriately allocated, causing frequent occurrence of the GC process.

Procedure

Check the GC duration.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **GC Duration Exceeds the Threshold**, and view the **Location** information. Check the name of the host for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the target cluster* > **Services** > **Flume**. On the page that is displayed, click the **Instance** tab. On the displayed tab page, select the role corresponding to the host name for which the alarm is generated and select **Customize** from the drop-down list in the upper right corner of the chart area. Choose **Agent** and select **Garbage Collection (GC) Duration of Flume**. Then, click **OK**.

Step 3 Check whether the GC duration of the Flume process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Flume** > **Configuration**. On the page that is displayed, click **All Configurations** and choose **Flume** > **System**. Set **-Xmx** in the **GC_OPTS** parameter to a larger value based on site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for the Flume server is insufficient for data transmission. You are advised to change the heap memory to: Channel capacity x Maximum size of a single data record x Number of channels. Note that the value of **xmx** cannot exceed the remaining memory of the node.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Flume** for the target cluster.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.169 ALM-24010 Flume Certificate File Is Invalid or Damaged

Description

Flume checks whether the Flume certificate file is valid (whether the certificate exists and whether the certificate format is correct) every hour. This alarm is generated when the certificate file is invalid or damaged. This alarm is automatically cleared when the certificate file becomes valid again.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24010	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Flume client cannot access the Flume server.

Possible Causes

The Flume certificate file is invalid or damaged.

Procedure

View alarm information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **ALM-24010 Flume Certificate File Is Invalid or Damaged**, and view the **Location** information. View the IP address of the instance for which the alarm is generated.

Check whether the certificate file in the system is valid. If it is not, generate a new one.

Step 2 Log in to the node for which the alarm is generated as user **root** and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to go to the Flume service certificate directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/conf

Step 4 Run the **ls -l** command to check whether the **flume_sChat.crt** file exists.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Run the **openssl x509 -in flume_sChat.crt -text -noout** command to check whether certificate details are displayed properly.

- If yes, go to **Step 9**.
- If no, go to **Step 6**.

Step 6 Run the following command to go to the Flume script directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/bin

Step 7 Run the following command to generate a new certificate. Then check whether the alarm is automatically cleared one hour later.

sh geneJKS.sh -f *Custom certificate password of the Flume role on the server* **-g** *Custom certificate password of the Flume role on the client*

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

  NOTE

  The custom certificate passwords must meet the following complexity requirements:

  - Contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
  - Contain 8 to 64 characters.
  - Be changed periodically (for example, every three months), and certificates and trust lists are generated again to ensure security.

Step 8 Check whether this alarm is generated again during periodic system check.

- If yes, go to **Step 9**.
- If no, no further action is required.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Select **Flume** in the required cluster for **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.170 ALM-24011 Flume Certificate File Is About to Expire

Description

Flume checks whether the Flume certificate file is about to expire every hour. This alarm is generated when the remaining validity period is at most 30 days. This alarm is automatically cleared when the remaining validity period is greater than 30 days.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24011	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Currently, there is no impact on the system.

Possible Causes

The Flume certificate file is about to expire.

Procedure

View alarm information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **ALM-24011 Flume Certificate Is About to Expire**, and view the **Location** information. View the IP address of the instance for which the alarm is generated.

Check whether the certificate file in the system is valid. If it is not, generate a new one.

Step 2 Log in to the node for which the alarm is generated as user **root** and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to go to the Flume service certificate directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/conf

Step 4 Run the following command to check the effective time and expiration time of the Flume user certificate:

openssl x509 -noout -text -in flume_sChat.crt

Step 5 Perform **Step 6** to **Step 7** during off-peak hours to update the certificate file as needed.

Step 6 Run the following command to go to the Flume script directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/bin

Step 7 Run the following command to generate a new certificate file. Then check whether the alarm is automatically cleared one hour later.

sh geneJKS.sh -f *Custom certificate password of the Flume role on the server* **-g** *Custom certificate password of the Flume role on the client*

- If yes, go to **Step 9**.
- If no, go to **Step 8**.

NOTE

The custom certificate passwords must meet the following complexity requirements:

- Contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
- Contain 8 to 64 characters.
- Be changed periodically (for example, every three months), and certificates and trust lists are generated again to ensure security.

Step 8 Log in to the Flume node for which the alarm is generated as user **omm** and repeat **Step 6** to **Step 7**. Then, check whether the alarm is automatically cleared one hour later.

- If yes, go to **Step 9**.
- If no, go to **Step 10**.

Step 9 Check whether this alarm is generated again during periodic system check.

- If yes, go to **Step 10**.
- If no, no further action is required.

Collect the fault information.

Step 10 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 11 Select **Flume** in the required cluster for **Service**.

Step 12 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.171 ALM-24012 Flume Certificate File Has Expired

Description

Flume checks whether its certificate file in the system has expired every hour. This alarm is generated when the server certificate has expired. This alarm is automatically cleared when the Flume certificate file becomes valid again.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24012	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Flume client cannot access the Flume server.

Possible Causes

The Flume certificate file has expired.

Procedure

View alarm information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **ALM-24012 Flume Certificate Has Expired**, and view the **Location** information. View the IP address of the instance for which the alarm is generated.

Check whether the certificate file in the system is valid. If it is not, generate a new one.

Step 2 Log in to the node for which the alarm is generated as user **root** and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to go to the Flume service certificate directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/conf

Step 4 Run the following command to check the effective time and expiration time of the HA user certificate to determine whether the certificate file is still in the validity period:

openssl x509 -noout -text -in flume_sChat.crt

- If yes, go to **Step 9**.
- If no, go to **Step 5**.

Step 5 Run the following command to go to the Flume script directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/bin

Step 6 Run the following command to generate a new certificate file. Then check whether the alarm is automatically cleared one hour later.

sh geneJKS.sh -f *Custom certificate password of the Flume role on the server* **-g** *Custom certificate password of the Flume role on the client*

- If yes, go to **Step 8**.
- If no, go to **Step 7**.

NOTE

The custom certificate passwords must meet the following complexity requirements:

- Contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
- Contain 8 to 64 characters.
- Be changed periodically (for example, every three months), and certificates and trust lists are generated again to ensure security.

Step 7 Log in to the Flume node for which the alarm is generated as user **omm** and repeat **Step 5** to **Step 6**. Then, check whether the alarm is automatically cleared one hour later.

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Check whether this alarm is generated again during periodic system check.

- If yes, go to **Step 9**.
- If no, no further action is required.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Select **Flume** in the required cluster for **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.172 ALM-24013 Flume MonitorServer Certificate File Is Invalid or Damaged

Description

MonitorServer checks whether its certificate file is valid (whether the certificate exists and whether the certificate format is correct) every hour. This alarm is generated when the certificate file is invalid or damaged. This alarm is automatically cleared when the certificate file becomes valid again.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24013	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Flume client cannot access the Flume server.

Possible Causes

The MonitorServer certificate file is invalid or damaged.

Procedure

View alarm information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row

containing **ALM-24013 MonitorServer Certificate File Is Invalid or Damaged**, and view the **Location** information. View the IP address of the instance for which the alarm is generated.

Check whether the certificate file in the system is valid. If it is not, generate a new one.

Step 2 Log in to the node for which the alarm is generated as user **root** and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to go to the MonitorServer certificate file directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/conf

Step 4 Run the **ls -l** command to check whether the **ms_sChat.crt** file exists:

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Run the **openssl x509 -in ms_sChat.crt -text -noout** command to check whether certificate details are displayed.

- If yes, go to **Step 9**.
- If no, go to **Step 6**.

Step 6 Run the following command to go to the Flume script directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/bin

Step 7 Run the following command to generate a new certificate file. Then check whether the alarm is automatically cleared one hour later.

sh geneJKS.sh -m *Custom password of the MonitorServer certificate on the server* **-n** *Custom password of the MonitorServer certificate on the client*

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

  NOTE

  The custom certificate passwords must meet the following complexity requirements:

  - Contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
  - Contain 8 to 64 characters.
  - Be changed periodically (for example, every three months), and certificates and trust lists are generated again to ensure security.

Step 8 Check whether this alarm is generated again during periodic system check.

- If yes, go to **Step 9**.
- If no, no further action is required.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Select **MonitorServer** in the required cluster for **Service**.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.173 ALM-24014 Flume MonitorServer Certificate Is About to Expire

Description

MonitorServer checks whether its certificate file is about to expire every hour. This alarm is generated when the remaining validity period is at most 30 days. This alarm is automatically cleared when the remaining validity period is greater than 30 days.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24014	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Currently, there is no impact on the system.

Possible Causes

The MonitorServer certificate file is about to expire.

Procedure

View alarm information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **ALM-24014 MonitorServer Certificate Is About to Expire**, and view the **Location** information. View the IP address of the instance for which the alarm is generated.

Check whether the certificate file in the system is valid. If it is not, generate a new one.

Step 2 Log in to the node for which the alarm is generated as user **root** and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to go to the MonitorServer certificate file directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/conf

Step 4 Run the following command to check the effective time and expiration time of the MonitorServer user certificate:

openssl x509 -noout -text -in ms_sChat.crt

Step 5 Perform **Step 6** to **Step 7** during off-peak hours to update the certificate file as needed.

Step 6 Run the following command to go to the Flume script directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/bin

Step 7 Run the following command to generate a new certificate file. Then check whether the alarm is automatically cleared one hour later.

sh geneJKS.sh -m *Custom password of the MonitorServer certificate on the server* **-n** *Custom password of the MonitorServer certificate on the client*

- If yes, go to **Step 9**.
- If no, go to **Step 8**.

  NOTE

  The custom certificate passwords must meet the following complexity requirements:
  - Contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
  - Contain 8 to 64 characters.
  - Be changed periodically (for example, every three months), and certificates and trust lists are generated again to ensure security.

Step 8 Log in to the Flume node for which the alarm is generated as user **omm** and repeat **Step 6** to **Step 7**. Then, check whether the alarm is automatically cleared one hour later.

- If yes, go to **Step 9**.
- If no, go to **Step 10**.

Step 9 Check whether this alarm is generated again during periodic system check.

- If yes, go to **Step 10**.
- If no, no further action is required.

Collect the fault information.

Step 10 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 11 Select **MonitorServer** in the required cluster for **Service**.

Step 12 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.174 ALM-24015 Flume MonitorServer Certificate File Has Expired

Description

MonitorServer checks whether its certificate file in the system has expired every hour. This alarm is generated when the server certificate has expired. This alarm is automatically cleared when the MonitorServer certificate file becomes valid again.

Attribute

Alarm ID	Alarm Severity	Auto Clear
24015	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The Flume client cannot access the Flume server.

Possible Causes

The MonitorServer certificate file has expired.

Procedure

View alarm information.

Step 1 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Alarm** > **Alarms**. On the page that is displayed, locate the row containing **ALM-24015 MonitorServer Certificate Has Expired**, and view the **Location** information. View the IP address of the instance for which the alarm is generated.

Check whether the certificate file in the system is valid. If it is not, generate a new one.

Step 2 Log in to the node for which the alarm is generated as user **root** and run the **su - omm** command to switch to user **omm**.

Step 3 Run the following command to go to the MonitorServer certificate file directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/conf

Step 4 Run the following command to check the effective time and expiration time of the user certificate to determine whether the certificate file is still in the validity period:

openssl x509 -noout -text -in ms_sChat.crt

- If yes, go to **Step 9**.
- If no, go to **Step 5**.

Step 5 Run the following command to go to the Flume script directory:

cd ${BIGDATA_HOME}/FusionInsight_Porter_*/install/FusionInsight-Flume-*/flume/bin

Step 6 Run the following command to generate a new certificate file. Then check whether the alarm is automatically cleared one hour later.

sh geneJKS.sh -m *Custom password of the MonitorServer certificate on the server* **-n** *Custom password of the MonitorServer certificate on the client*

- If yes, go to **Step 8**.
- If no, go to **Step 7**.

NOTE

The custom certificate passwords must meet the following complexity requirements:

- Contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
- Contain 8 to 64 characters.
- Be changed periodically (for example, every three months), and certificates and trust lists are generated again to ensure security.

Step 7 Log in to the Flume node for which the alarm is generated as user **omm** and repeat **Step 5** to **Step 6**. Then, check whether the alarm is automatically cleared one hour later.

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Check whether this alarm is generated again during periodic system check.

- If yes, go to **Step 9**.
- If no, no further action is required.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Select **MonitorServer** in the required cluster for **Service**.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.175 ALM-25000 LdapServer Service Unavailable

Description

The system checks the LdapServer service status every 30 seconds. This alarm is generated when the system detects that both the active and standby LdapServer services are abnormal.

This alarm is cleared when the system detects that one or two LdapServer services are normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
25000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

When this alarm is generated, no operation can be performed for the KrbServer users and LdapServer users in the cluster. For example, users, user groups, or roles cannot be added, deleted, or modified, and user passwords cannot be changed on the FusionInsight Manager portal. The authentication for existing users in the cluster is not affected.

Possible Causes

- The node where the LdapServer service locates is faulty.
- The LdapServer process is abnormal.

Procedure

Check whether the nodes where the two SlapdServer instances of the LdapServer service are located are faulty.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **LdapServer** > **Instance** to go to the LdapServer instance page to obtain the host name of the node where the two SlapdServer instances locates.

Step 2 Choose **O&M** > **Alarm** > **Alarms**. On the **Alarm** page of the FusionInsight Manager system, check whether any alarm of **Node Fault** exists.

- If yes, go to **Step 3**.
- If no, go to **Step 6**.

Step 3 Check whether the host name in the alarm is consistent with the **Step 1** host name.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 Handle the alarm according to "ALM-12006 Node Fault".

Step 5 Check whether **LdapServer Service Unavailable** is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 10**.

Check whether the LdapServer process is normal.

Step 6 Choose **O&M** > **Alarm** > **Alarms**. On the **Alarm** page of the FusionInsight Manager system, check whether any alarm of **Process Fault** exists.

- If yes, go to **Step 7**.
- If no, go to **Step 10**.

Step 7 Check whether the service and host name in the alarm are consistent with the LdapServer service and host name.

- If yes, go to **Step 8**.
- If no, go to **Step 10**.

Step 8 Handle the alarm according to "ALM-12007 Process Fault".

Step 9 Check whether **LdapServer Service Unavailable** is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 10**.

Collect fault information.

Step 10 On the FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 11 Select **LdapServer** in the required cluster from the **Service**.

Step 12 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.176 ALM-25004 Abnormal LdapServer Data Synchronization

Description

The system checks the LdapServer data every 30 seconds. This alarm is generated when the data on the active and standby LdapServers of Manager is inconsistent for 12 consecutive times. This alarm is cleared when the data on the active and standby LdapServers is consistent.

The system checks the LdapServer data every 30 seconds. This alarm is generated when the LdapServer data in the cluster is inconsistent with that on Manager for 12 consecutive times. This alarm is cleared when the data is consistent.

Attribute

Alarm ID	Alarm Severity	Auto Clear
25004	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

LdapServer data inconsistency occurs because the LdapServer data in Manager is damaged or the LdapServer data in the cluster is damaged. The LdapServer process with damaged data cannot provide services externally, and the authentication functions of Manager and the cluster are affected.

Possible Causes

- The network of the node where the LdapServer process locates is faulty.

- The LdapServer process is abnormal.
- The OS restart damages data on LdapServer.

Procedure

Check whether the network where the LdapServer nodes reside is faulty.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. Record the IP address of HostName in the alarm locating information as IP1 (if multiple alarms exist, record the IP addresses as IP1, IP2, and IP3 respectively).

Step 2 Contact O&M personnel and log in to the nodes corresponding to IP 1. Run the ping command to check whether the IP address of the management plane of the active OMS node can be pinged.

- If yes, go to **Step 4**.
- If no, go to **Step 3**.

Step 3 Contact the network administrator to recover the network and check whether **Abnormal LdapServer Data Synchronization** is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Check whether the LdapServer processes are normal.

Step 4 On the **Alarm** page of FusionInsight Manager, check whether the **OLdap Resource Abnormal** exists.

- If yes, go to **Step 5**.
- If no, go to **Step 7**.

Step 5 Clear the alarm by following the steps provided in "ALM-12004 OLdap Resource Abnormal".

Step 6 Check whether **Abnormal LdapServer Data Synchronization** is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 7**.

Step 7 On the **Alarm** page of FusionInsight Manager, check whether **Process Faul**t is generated for the LdapServer service.

- If yes, go to **Step 8**.
- If no, go to **Step 10**.

Step 8 Handle the alarm according to "ALM-12007 Process Fault".

Step 9 Check whether **Abnormal LdapServer Data Synchronization** is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

Check whether the LdapServer processes are normal.

Step 10 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Record the IP address of HostName in the alarm locating information as "IP1" (if multiple alarms exist, record the IP addresses as "IP1", "IP2", and "IP3" respectively). Choose **Cluster** >

Name of the desired cluster > **Services** > **LdapServer** > **Configurations**. Record the port number of LdapServer as "PORT". (If the IP address in the alarm locating information is the IP address of the standby management node, choose **System** > **OMS** > **oldap** > **Modify Configuration** and record the listening port number of LdapServer.)

Step 11 Log in to the nodes corresponding to IP1 as user **omm**.

Step 12 Run the following command to check whether errors are displayed in the queried information.

ldapsearch -H ldaps://*IP1*:*PORT* **-LLL -x -D cn=root,dc=hadoop,dc=com -W -b ou=Peoples,dc=hadoop,dc=com**

After running the command, enter the **LDAP** administrator password. Contact the system administrator to obtain the password.

- If yes, go to **Step 13**.
- If no, go to **Step 15**.

Step 13 Recover the LdapServer and OMS nodes using data backed up before the alarm is generated.

NOTE

Use the OMS data and LdapServer data backed up at the same point in time to recover the data. Otherwise, the service and operation may fail. To recover data when services run properly, you are advised to manually back up the latest management data and then recover the data. Otherwise, Manager data produced between the backup point in time and the recovery point in time will be lost.

Step 14 Check whether alarm **Abnormal LdapServer Data Synchronization** is cleared.

- If yes, no further action is required.
- If no, go to **Step 15**.

Collect fault information.

Step 15 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 16 Select **LdapServer** in the required cluster and **OmsLdapServer** from the **Service**.

Step 17 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.177 ALM-25005 nscd Service Exception

Description

The system checks the status of the nscd service every 60 seconds. This alarm is generated when the nscd process fails to be queried for four consecutive times (three minutes) or users in LdapServer cannot be obtained.

This alarm is cleared when the process is restored and users in LdapServer can be obtained.

Attribute

Alarm ID	Alarm Severity	Auto Clear
25005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

The alarmed node may not be able to synchronize data from LdapServer. The **id** command may fail to obtain the LDAP data, affecting upper-layer services.

Possible Causes

- The nscd service is not started.
- The network is faulty and cannot access the LDAP server.
- NameService is abnormal.
- Users cannot be queried because the OS executes commands too slowly.

Procedure

Check whether the nscd service is started.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. Find the IP address of **HostName** in **Location** of the alarm and record it as IP1 (if multiple alarms exist, record the IP addresses as IP1, IP2, and IP3 respectively).

Step 2 Contact the O&M personnel to access the node using IP1 as user root. Run the **ps -ef | grep nscd** command and check whether the **/usr/sbin/nscd** process is started.

- If the process is started, go to **Step 5**.
- If the process is not started, go to **Step 3**.

Step 3 Run the **service nscd restart** command as user **root** to restart the nscd service. Then run the **ps -ef | grep nscd** command to check whether the nscd service is started.

- If it is started, go to **Step 4**.
- If it is not started, go to **Step 15**.

Step 4 Wait for 5 minutes and run the **ps -ef | grep nscd** command again as user **root**. Check whether the service exists.

- If it exists, go to **Step 11**.
- If it does not exist, go to **Step 15**.

Check whether the LDAP server can be accessed.

Step 5 Log in to the alarmed node as user **root**. Run the **ping** command to check the network connectivity between this node and the LdapServer node.

- If the network is normal, go to **Step 6**.
- If the network is faulty, contact network administrators to troubleshoot the fault.

Check whether NameService is normal.

Step 6 Log in to the alarmed node as user **root**. Run the **cat /etc/nsswitch.conf** command and check the **passwd**, **group**, **services**, **netgroup**, and **aliases** configurations of NameService.

The correct parameter configurations are as follows: **passwd: compat ldap**, **group: compat ldap**, **services: files ldap**, **netgroup: files ldap**, and **aliases: files ldap**.

- If the configurations are correct, go to **Step 7**.
- If the configurations are incorrect, go to **Step 9**.

Step 7 Log in to the alarmed node as user **root**. Run the **cat /etc/nscd.conf** command and check the **enable-cache passwd**,**positive-time-to-live passwd**,**enable-cache group** and **positive-time-to-live group** configurations of NameService.

The correct parameter configurations are as follows: **enable-cache passwd yes**, **positive-time-to-live passwd 600**, **enable-cache group yes** and **positive-time-to-live group 3600**.

- If the configurations are correct, go to **Step 8**.
- If the configurations are incorrect, go to **Step 10**.

Step 8 Run the **/usr/sbin/nscd -i group** and **/usr/sbin/nscd -i passwd** commands as user **root**. Wait for 2 minutes and run the **id admin** and **id backup/manager** commands to check whether results can be queried.

- If results are queried, go to **Step 11**.

- If no result is queried, go to **Step 15**.

Step 9 Run the **vi /etc/nsswitch.conf** command as user **root**. Correct the configurations in **Step 6** and save the file. Run the **service nscd restart** command to restart the nscd service. Wait for 2 minutes and run the **id admin** and **id backup/manager** commands to check whether results can be queried.

- If results are queried, go to **Step 11**.
- If no result is queried, go to **Step 15**.

Step 10 Run the **vi /etc/nscd.conf** command as user **root**. Correct the configurations in **Step 7** and save the file. Run the **service nscd restart** command to restart the nscd service. Wait for 2 minutes and run the **id admin** and **id backup/manager** commands to check whether results can be queried.

- If results are queried, go to **Step 11**.
- If no result is queried, go to **Step 15**.

Step 11 Log in to the FusionInsight Manager portal. Wait for 5 minutes and check whether the **nscd Service Exception** alarm is cleared.

- If the alarm is cleared, no further action is required.
- If the alarm persists, go to **Step 12**.

Check whether frame freezing occurs when running a command in the operating system.

Step 12 Log in to the faulty node as user **root**, run the **id admin** command, and check whether the command execution takes a long time. If the command execution takes more than 3 seconds, the command execution is deemed to be slow.

- If yes, go to **Step 13**.
- If no, go to **Step 15**.

Step 13 Run the **cat /var/log/messages** command to check whether the nscd frequently restarts or the error information **Can't contact LDAP server** exists.

nscd exception example:

```
Feb 11 11:44:42 10-120-205-33 nscd: nss_ldap: failed to bind to LDAP server ldaps://10.120.205.55:21780: Can't contact LDAP server
Feb 11 11:44:43 10-120-205-33 ntpq: nss_ldap: failed to bind to LDAP server ldaps://10.120.205.55:21780: Can't contact LDAP server
Feb 11 11:44:44 10-120-205-33 ntpq: nss_ldap: failed to bind to LDAP server ldaps://10.120.205.92:21780: Can't contact LDAP server
```

- If yes, go to **Step 14**.
- If no, go to **Step 15**.

Step 14 Run the **vi $BIGDATA_HOME/tmp/random_ldap_ip_order** command to modify the number at the end. If the original number is an odd number, change it to an even number. If the number is an even number, change it to an odd number.

Run the **vi /etc/ldap.conf** command to enter the editing mode, press **Insert** to start editing, and then change the first two IP addresses of the URI configuration item.

After the modification is complete, press **Esc** to exit the editing mode and enter **:wq** to save the settings and exit.

Run the **service nscd restart** command to restart the nscd service. Wait 5 minutes and run the **id admin** command again. Check whether the command execution is slow.

- If yes, go to **Step 15**.
- If no, log in to other faulty nodes and run **Step 12** to **Step 14** and check whether the first ldapserver node in the URI before modifying **/etc/ldap.conf** is faulty. For example, check whether the service IP address is unreachable, the network delay is too long, or other abnormal software is deployed.

Collect fault information.

Step 15 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 16 Select **LdapClient** in the required cluster from the **Service**.

Step 17 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 18 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.178 ALM-25006 Sssd Service Exception

Description

The system checks the status of the sssd service every 60 seconds. This alarm is generated when the sssd process fails to be queried for four consecutive times (three minutes) or users in LdapServer cannot be obtained.

This alarm is cleared when the process is restored and users in LdapServer can be obtained.

Attribute

Alarm ID	Alarm Severity	Auto Clear
25006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.

Impact on the System

The alarmed node may not be able to synchronize data from LdapServer. The id command may fail to obtain the LDAP data, affecting upper-layer services.

Possible Causes

- The sssd service is not started or is incorrectly started.
- The network is faulty and cannot access the LDAP server.
- NameService is abnormal.
- Users cannot be queried because the OS executes commands too slowly.

Procedure

Check whether the sssd service is correctly started.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms**. Find the IP address of **HostName** in **Location** of the alarm and record it as IP1 (if multiple alarms exist, record the IP addresses as IP1, IP2, and IP3 respectively).

Step 2 Contact the O&M personnel to access the node using IP1 as user root. Run the **ps -ef | grep sssd** command and check whether the **/usr/sbin/sssd** process is started.

- If the process is started, go to **Step 3**.
- If the process is not started, go to **Step 4**.

Step 3 Check whether the sssd process queried in **Step 2** has three subprocesses.

- If yes, go to **Step 5**.
- If no, go to **Step 4**.

Step 4 Run the **service sssd restart** command as user **roo**t to restart the sssd service. Then run the **ps -ef | grep sssd** command to check whether the sssd process is normal.

In the normal state, the **/usr/sbin/sssd** process has three subprocesses: **/usr/libexec/sssd/sssd_be**, **/usr/libexec/sssd/sssd_nss**, and **/usr/libexec/sssd/sssd_pam**.

- If it exists, go to **Step 9**.

- If it does not exist, go to **Step 13**.

Check whether the LDAP server can be accessed.

Step 5 Log in to the alarmed node as user **root**. Run the **ping** command to check the network connectivity between this node and the LdapServer node.

- If the network is normal, go to **Step 6**.
- If the network is faulty, contact network administrators to troubleshoot the fault.

Check whether NameService is normal.

Step 6 Log in to the alarmed node as user **root**. Run the **cat /etc/nsswitch.conf** command and check the **passwd** and **group** configurations of NameService.

The correct parameter configurations are as follows: **passwd: compat ldap** and **group: compat ldap**.

- If the configurations are correct, go to **Step 7**.
- If the configurations are incorrect, go to **Step 8**.

Step 7 Run the **/usr/sbin/sss_cache -G** and **/usr/sbin/sss_cache -U** commands as user **root**. Wait for 2 minutes and run the **id admin** and **id backup/manager** commands to check whether results can be queried.

- If results are queried, go to **Step 9**.
- If no result is queried, go to **Step 13**.

Step 8 Run the **vi /etc/nsswitch.conf** command as user **root**. Correct the configurations in **Step 6** and save the file. Run the **service sssd restart** command to restart the sssd service. Wait for 2 minutes and run the **id admin** and **id backup/manager** commands to check whether results can be queried.

- If results are queried, go to **Step 9**.
- If no result is queried, go to **Step 13**.

Step 9 Log in to the FusionInsight Manager portal. Wait for 5 minutes and check whether the **sssd Service Exception** alarm is cleared.

- If the alarm is cleared, no further action is required.
- If the alarm persists, go to **Step 10**.

Check whether frame freezing occurs when running a command in the operating system.

Step 10 Log in to the faulty node as user **root**, run the **id admin** command, and check whether the command execution takes a long time. If the command execution takes more than 3 seconds, the command execution is deemed to be slow.

- If yes, go to **Step 11**.
- If no, go to **Step 13**.

Step 11 Run the **cat /var/log/messages** command to check whether the sssd frequently restarts or the error information **Can't contact LDAP server** exists.

sssd restart example:

```
Feb  7 11:38:16 10-132-190-105 sssd[pam]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd[nss]: Shutting down
```

```
Feb  7 11:38:16 10-132-190-105 sssd[nss]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd[be[default]]: Shutting down
Feb  7 11:38:16 10-132-190-105 sssd: Starting up
Feb  7 11:38:16 10-132-190-105 sssd[be[default]]: Starting up
Feb  7 11:38:16 10-132-190-105 sssd[nss]: Starting up
Feb  7 11:38:16 10-132-190-105 sssd[pam]: Starting up
```

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 Run the **vi $BIGDATA_HOME/tmp/random_ldap_ip_order** command to modify the number at the end. If the original number is an odd number, change it to an even number. If the number is an even number, change it to an odd number.

Run the **vi /etc/sssd/sssd.conf** command to reverse the first two IP addresses of the **ldap_uri** configuration item, save the settings, and exit.

Run the **ps -ef | grep sssd** command to query the ID of the sssd process, kill it, and run the **/usr/sbin/sssd -D -f** command to restart the sssd service. Wait 5 minutes and run the **id admin** command again.

Check whether the command execution is slow.

- If yes, go to **Step 13**.
- If no, log in to other faulty nodes and run **Step 10** to **Step 12**. Collect logs and check whether the first ldapserver node in the ldap_uri before modifying **/etc/sssd/sssd.conf** is faulty. For example, check whether the service IP address is unreachable, the network latency is too long, or other abnormal software is deployed.

Collect fault information.

Step 13 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 14 Select **LdapClient** in the required cluster from the **Service**.

Step 15 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.179 ALM-25500 KrbServer Service Unavailable

Description

The system checks the KrbServer service status every 30 seconds. This alarm is generated when the system detects that the KrbServer service is abnormal.

This alarm is cleared when the system detects that the KrbServer service is normal.

Attribute

Alarm ID	Alarm Severity	Auto Clear
25500	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

When this alarm is generated, no operation can be performed for the KrbServer component in the cluster. The authentication of KrbServer in other components will be affected. The running status of components that depend on KrbServer in the cluster is Bad.

Possible Causes

- The node where the KrbServer service locates is faulty.
- The OLdap service is abnormal.

Procedure

Check whether the node where the KrbServer service locates is faulty.

Step 1 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **KrbServer** > **Instance** to go to the KrbServer instance page to obtain the host name of the node where the KrbServer service locates.

Step 2 On the **Alarm** page of the FusionInsight Manager system, check whether any alarm of **Node Fault** exists.

- If yes, go to **Step 3**.
- If no, go to **Step 6**.

Step 3 Check whether the host name in the alarm is consistent with the **Step 1** host name.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 Handle the alarm according to "ALM-12006 Node Fault".

Step 5 Check whether **KrbServer Service Unavailable** is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check whether the OLdap service is normal.

Step 6 On the **Alarm** page of the FusionInsight Manager system, check whether any alarm of **OLdap Resource Abnormal** exists.

- If yes, go to **Step 7**.
- If no, go to **Step 9**.

Step 7 Handle the alarm according to "ALM-12004 OLdap Resource Abnormal".

Step 8 Check whether **KrbServer Service Unavailable** is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On the FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 Select **KrbServer** in the required cluster from the **Service**.

Step 11 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.180 ALM-26051 Storm Service Unavailable

Description

The system checks the Storm service status every 30 seconds. This alarm is generated when all Nimbus nodes in the cluster are abnormal and the Storm service is unavailable.

This alarm is cleared when the Storm service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
26051	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The cluster cannot provide the Storm service, and users cannot perform new Storm tasks.

Possible Causes

- The Kerberos cluster is faulty.
- The ZooKeeper cluster is faulty or suspended.
- The active and standby Nimbus nodes in the Storm cluster are abnormal

Procedure

Check the status of the Kerberos cluster. (Skip this step if the normal mode is used.)

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 2 Check whether the running status of the Kerberos service is **Normal**.

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 See the related maintenance information of **ALM-25500 KrbServer Service Unavailable**.

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 5**.

Check the status of the ZooKeeper cluster.

Step 5 Check whether the running status of the ZooKeeper service is **Normal**.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 If ZooKeeper service is stopped, start it, else see the related maintenance information of **ALM-13000 ZooKeeper Service Unavailable**.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Check the status of the active and standby Nimbus nodes.

Step 8 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Nimbus** to go to the Nimbus Instances page.

Step 9 Check whether only one Nimbus node that is in the **Active** state in **Roles**.

- If yes, go to **Step 13**.
- If no, go to **Step 10**.

Step 10 Select two Nimbus role instances, choose **More** > **Restart Instance**, and check whether the instances restart successfully.

- If yes, go to **Step 11**.
- If no, go to **Step 13**.

Step 11 Log in to the FusionInsight Manager portal again, choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Nimbus** to check whether the running status is **Normal**.

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 Wait for 30 seconds and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collecting Fault Information

Step 13 On the FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 14 Select the following nodes in the required cluster from the **Service** drop-down list:

- KrbServer

  NOTE

  KrbServer logs do not need to be downloaded in normal mode.

- ZooKeeper
- Storm

Step 15 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.181 ALM-26052 Number of Available Supervisors of the Storm Service Is Less Than the Threshold

Description

The system periodically checks the number of available Supervisors every 60 seconds and compares the number of available Supervisors with the threshold. This alarm is generated when the number of available Supervisors is less than the threshold.

You can change the threshold in **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster*.

This alarm is cleared when the number of available Supervisors is greater than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
26052	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Existing tasks in the cluster cannot be performed. The cluster can receive new Storm tasks, but cannot perform these tasks.

Possible Causes

The status of some Supervisors in the cluster is abnormal.

Procedure

Check the Supervisor status.

Step 1 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Supervisor** to go to the Storm service management page.

Step 2 In **Roles**, check whether any instance whose status is **Faulty** or **Restoring** exists.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Select Supervisor role instances whose status is **Faulty** or **Restoring**, choose **More** > **Restart Instance**, and check whether the instances restart successfully.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Wait for 30 seconds, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

NOTE

Services are interrupted when the Supervisor is being restarted. Then, services are restored after the restarting.

Collect fault information.

Step 5 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 6 Select **Storm** and **ZooKeeper** in the required cluster from the **Service** drop-down list box.

Step 7 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 8 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.182 ALM-26053 Storm Slot Usage Exceeds the Threshold

Description

The system checks the slot usage every 60 seconds and compares the actual slot usage with the threshold. This alarm is generated when the slot usage is greater than the threshold.

You can change the threshold in **O&M** > **Alarm** > **Thresholds**.

This alarm is cleared when the slot usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
26053	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
Trigger condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

New Storm tasks cannot be performed.

Possible Causes

- The status of some Supervisors in the cluster is abnormal.
- The status of all Supervisors is normal, but the processing capability is insufficient.

Procedure

Check the Supervisor status.

Step 1 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Instance** to go to the Storm instance management page.

Step 2 Check whether any instance whose status is **Faulty** or **Restoring** exists.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Select Supervisor role instances whose status is **Faulty** or **Restoring**, choose **More** > **Restart Instance**, and check whether the instances restart successfully.

- If yes, go to **Step 4**.
- If no, go to **Step 10**.

Step 4 Wait several minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Increase the number of slots in each Supervisor.

Step 5 Log in to the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Configurations** > **All Configurations**.

Step 6 Increase the number of ports in the **supervisor.slots.ports** parameter of each Supervisor role and restart the instance.

Step 7 Wait several minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Step 8 Perform capacity expansion for Supervisor.

Step 9 Wait several minutes, and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 10**.

NOTE

Services are interrupted when the Supervisor is being restarted. Then, services are restored after the restarting.

Collect fault information.

Step 10 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 11 Select **Storm** and **ZooKeeper** in the required cluster from the **Service** drop-down list box.

Step 12 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.183 ALM-26054 Nimbus Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of Storm Nimbus every 30 seconds and compares the actual usage with the threshold. The alarm is generated when the heap memory usage of Storm Nimbus exceeds the threshold (80% of the maximum memory by default) for 5 consecutive times.

Users can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Storm** > **Nimbus** to change the threshold.

The alarm is cleared when the heap memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
26054	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

When the heap memory usage of Storm Nimbus is overhigh, frequent GCs occur. In addition, a memory overflow may occur so that the Yarn service is unavailable.

Possible Causes

The heap memory of the Storm Nimbus instance on the node is overused or the heap memory is inappropriately allocated. As a result, the usage exceeds the threshold.

Procedure

Check the heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **Heap Memory Usage of Storm Nimbus Exceeds the Threshold** > **Location**. Check the host name of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down menu in the chart area and choose **Customize** > **Nimbus** > **Heap Memory Usage of Nimbus**. Click **OK**.

Step 3 Check whether the used heap memory of Nimbus reaches the threshold (The default value is 80% of the maximum heap memory) specified for Nimbus.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Storm** > **Configurations** > **All Configurations** > **Nimbus** >

System. Change the value of **-Xmx** in **NIMBUS_GC_OPTS** based on site requirements, and click **Save**. Click **OK**.

NOTE

- You are advised to set **-Xms** and **-Xmx** to the same value to prevent adverse impact on performance when JVM dynamically adjusts the heap memory size.
- The number of Workers grows as the Storm cluster scale increases. You can increase the value of **GC_OPTS** for Nimbus. The recommended value is as follows: If the number of Workers is 20, set **-Xmx** to a value greater than or equal to 1 GB. If the number of Workers exceeds 100, set **-Xmx** to a value greater than or equal to 5 GB.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select the following node in the required cluster from the **Service** drop-down list.

- NodeAgent
- Storm

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.184 ALM-27001 DBService Service Unavailable

Description

The alarm module checks the DBService service status every 30 seconds. This alarm is generated when the system detects that DBService service is unavailable.

This alarm is cleared when DBService service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
27001	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The database service is unavailable and cannot provide data import and query functions for upper-layer services, which results in some services exceptions.

Possible Causes

- The floating IP address does not exist.
- There is no active DBServer instance.
- The active and standby DBServer processes are abnormal.

Procedure

Check whether the floating IP address exists in the cluster environment.

Step 1 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService** > **Instance**.

Step 2 Check whether the active instance exists.

- If yes, go to **Step 3**.
- If no, go to **Step 9**.

Step 3 Select the active DBServer instance and record the IP address.

Step 4 Log in to the host that corresponds to the preceding IP address as user **root**, and run the **ifconfig** command to check whether the DBService floating IP address exists on the node.

- If yes, go to **Step 5**.
- If no, go to **Step 9**.

Step 5 Run the **ping** *floatip* command to check whether the DBService floating IP address can be pinged successfully.

- If yes, go to **Step 6**.
- If no, go to **Step 9**.

Step 6 Log in to the host that corresponds to the DBService floating IP address as user **root**, and run the command to delete the floating IP address.

ifconfig *interface* **down**

Step 7 On the FusionInsight Manager home page, choose **Cluster >** *Name of the desired cluster* > **Services** > **DBService** > **More** > **Restart Service** to restart DBService, and check whether DBService is restarted successfully.

- If yes, go to **Step 8**.
- If no, go to **Step 9**.

Step 8 Wait for about 2 minutes and check whether the alarm is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 14**.

Check the status of the active DBServer instance.

Step 9 Select the DBServer instance whose role status is abnormal and record the IP address.

Step 10 On the **Alarm** page, check whether **Process Fault** occurs in the DBServer instance on the host that corresponds to the IP address.

- If yes, go to **Step 11**.
- If no, go to **Step 14**.

Step 11 Handle the alarm according to "ALM-12007 Process Fault".

Step 12 Wait for about 5 minutes and check whether the alarm is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 19**.

Check the status of the active and standby DBServers.

Step 13 Log in to the host that corresponds to the preceding IP address as user **root**, and run the **su - omm** command to switch to user **omm**.

Step 14 Run the **cd ${DBSERVER_HOME}** command to go to the installation directory of the DBService.

Step 15 Run the **sh sbin/status-dbserver.sh** command to view the status of the active and standby HA processes of DBService. Determine whether the status can be viewed successfully.

```
HAMode
double

NodeName            HostName        HAVersion          StartTime           HAActive
HAAllResOK          HARunPhase
10_5_89_12          host01          V100R001C01        2019-06-13 21:33:09 active
normal              Actived
10_5_89_66          host03          V100R001C01        2019-06-13 21:33:09 standby
normal              Deactived

NodeName            ResName         ResStatus          ResHAStatus         ResType
10_5_89_12          floatip         Normal             Normal              Single_active
10_5_89_12          gaussDB         Active_normal      Normal              Active_standby
10_5_89_66          floatip         Stopped            Normal              Single_active
10_5_89_66          gaussDB         Standby_normal     Normal              Active_standby
```

- If yes, go to **Step 16**.
- If no, go to **Step 19**.

Step 16 Check whether the active and standby HA processes are in the abnormal state.

- If yes, go to **Step 17**.
- If no, go to **Step 19**.

Step 17 On FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* > **Services** > **DBService** > **More** > **Restart Service** to restart DBService, and check whether the system displays a message indicating that the restart is successful.

- If yes, go to **Step 18**.
- If no, go to **Step 19**.

Step 18 Wait for about 2 minutes and check whether the alarm is cleared in the alarm list.

- If yes, no further action is required.
- If no, go to **Step 19**.

Collect fault information.

Step 19 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 20 Select **DBService** in the required cluster and **NodeAgent** from the **Service**.

Step 21 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 22 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.185 ALM-27003 DBService Heartbeat Interruption Between the Active and Standby Nodes

Description

This alarm is generated when the active or standby DBService node does not receive heartbeat messages from the peer node for 7 seconds.

This alarm is cleared when the heartbeat recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
27003	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Local DBService HA Name	Specifies a local DBService HA.
Peer DBService HA Name	Specifies a peer DBService HA.

Impact on the System

During the DBService heartbeat interruption, only one node can provide the service. If this node is faulty, no standby node is available for failover and the service is unavailable.

Possible Causes

The link between the active and standby DBService nodes is abnormal.

Procedure

Check whether the network between the active DBService server and the standby DBService server is normal.

Step 1 In the alarm list on FusionInsight Manager, click ⌄ in the row where the alarm is located in the real-time alarm list and view the standby DBService server address.

Step 2 Log in to the active DBService server as user **root**.

Step 3 Run the **ping** *standby DBService heartbeat IP address* command to check whether the standby DBService server is reachable.

- If yes, go to **Step 6**.
- If no, go to **Step 4**.

Step 4 Contact the network administrator to check whether the network is faulty.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Rectify the network fault and check whether the alarm is cleared from the alarm list.

- If yes, no further action is required.

- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select the following nodes in the required cluster from the **Service**:

- DBService
- Controller
- NodeAgent

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.186 ALM-27004 Data Inconsistency Between Active and Standby DBServices

Description

The system checks the data synchronization status between the active and standby DBService every 10 seconds. This alarm is generated when the synchronization status cannot be queried for six consecutive times or when the synchronization status is abnormal.

This alarm is cleared when the synchronization status becomes normal.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
27004	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Local DBService HA Name	Specifies the HA name of the local DBService.
Peer DBService HA Name	Specifies the HA name of the peer DBService.
SYNC_PERCENT	Specifies the synchronization percentage.

Impact on the System

When data is not synchronized between the active and standby DBServices, data may be lost or abnormal if the active instance becomes abnormal.

Possible Causes

- The network between the active and standby nodes is unstable.
- The standby DBService is abnormal.
- The standby node disk space is full.

Procedure

Check whether the network between the active and standby nodes is normal.

Step 1 On FusionInsight Manager, choose **Cluster > Services > DBService > Instance**, check the service IP address of the standby DBServer instance.

Step 2 Log in to the active DBService node as user **root**.

Step 3 Run the **ping** *Standby DBService heartbeat IP address* command to check whether the standby DBService node is reachable.

- If yes, go to **Step 6**.
- If no, go to **Step 4**.

Step 4 Contact the network administrator to check whether the network is faulty.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Rectify the network fault and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Check whether the standby DBService is normal.

Step 6 Log in to the standby DBService node as user **root**.

Step 7 Run the **su - omm** command to switch to user **omm**.

Step 8 Go to the **${DBSERVER_HOME}/sbin** directory and run the **./status-dbserver.sh** command to check whether the GaussDB resource status of the standby DBService is normal. In the command output, check whether the following information is displayed in the row where **ResName** is **gaussDB**:

For example:

```
10_10_10_231 gaussDB Standby_normal Normal Active_standby
```

- If yes, go to **Step 9**.
- If no, go to **Step 16**.

Check whether the standby node disk space is full.

Step 9 Log in to the standby DBService node as user **root**.

Step 10 Run the **su - omm** command to switch to user **omm**.

Step 11 Go to the **${DBSERVER_HOME}** directory, and run the following commands to obtain the DBService data directory:

cd ${DBSERVER_HOME}

source .dbservice_profile

echo ${DBSERVICE_DATA_DIR}

Step 12 Run the **df -h** command to view the system disk partition usage information.

Step 13 Check whether the DBService data directory space is full.

- If yes, go to **Step 14**.
- If no, go to **Step 16**.

Step 14 Expand the disk capacity.

Step 15 After the disk capacity is expanded, wait 2 minutes and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 16**.

Collect fault information.

Step 16 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 17 Select **DBService** in the required cluster from the **Service**.

Step 18 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 19 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.187 ALM-27005 Database Connections Usage Exceeds the Threshold

Description

The system checks the usage of the number of database connections of the nodes where DBServer instances are located every 30 seconds and compares the usage with the threshold. If the usage exceeds the threshold for five consecutive times (this number is configurable, and 5 is the default value), the system generates this alarm. The default usage threshold is 90%, and you can configure it based on site requirements.

The trigger count is configurable. This alarm is cleared in the following scenarios:

- The trigger count is 1, and the usage of the number of database connections is less than or equal to the threshold.
- The trigger count is greater than 1, and the usage of the number of database connections is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
27005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Upper-layer services may fail to connect to the DBService database, affecting services.

Possible Causes

- Too many database connections are used.
- The maximum number of database connections is improperly configured.
- The alarm threshold or alarm trigger count is improperly configured.

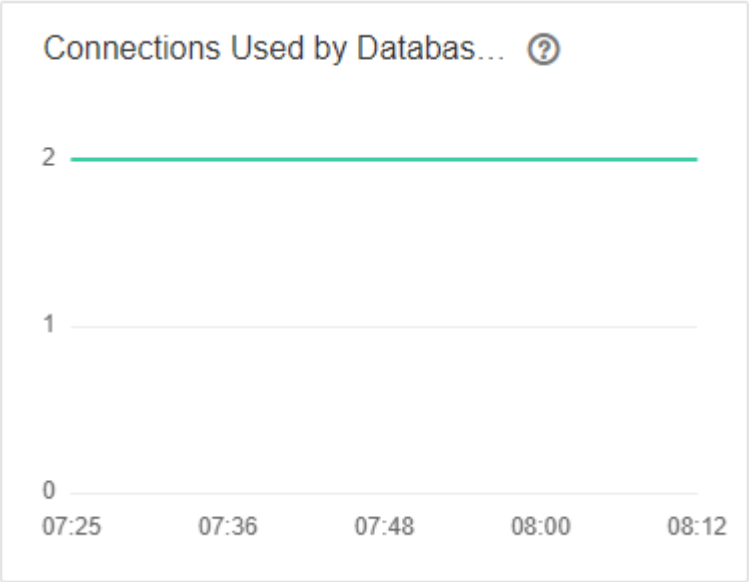
Procedure

Checking whether too many data connections are used

Step 1 On FusionInsight Manager, click DBService in the service list on the left navigation pane. The DBService monitoring page is displayed.

Step 2 Observe the number of connections used by the database user, as shown in **Figure 10-39**. Based on the service scenario, reduce the number of database user connections.

Figure 10-39 Number of connections used by database users



Step 3 Wait for 2 minutes and check whether the alarm is automatically cleared.

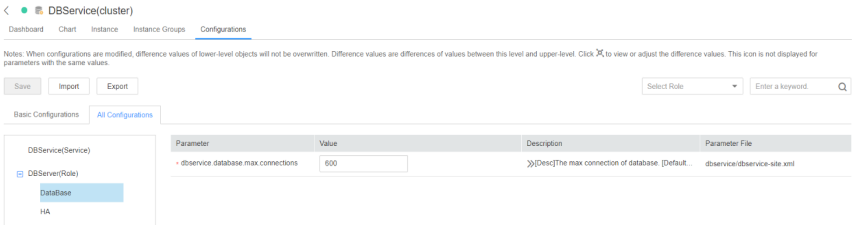
- If it is, no further action is required.

- If it is not, go to **Step 4**.

Checking whether the maximum number of database connections is properly configured

Step 4 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService** > **Configurations**. On the displayed page, select the **All Configurations** tab, and increase the maximum number of database connections based on service requirements, as shown in **Figure 10-40**. Click **Save**. In the displayed **Save configuration** dialog box, click **OK**.

Figure 10-40 Setting the maximum number of database connections



Step 5 After the maximum number of database connections is changed, restart DBService (do not restart the upper-layer services).

Procedure: Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService**. On the displayed page, choose **More** > **Restart Service**. Enter the password of the current login user and click **OK**. Do not select **Restart upper-layer services.**, click **OK**.

Step 6 After the service is restarted, wait for 2 minutes and check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 7**.

Checking whether the alarm threshold or trigger count is properly configured

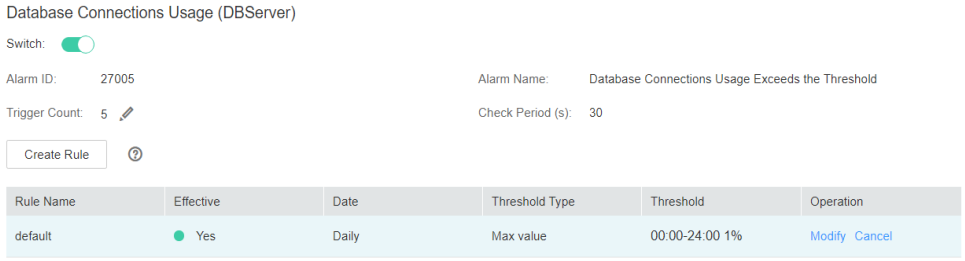
Step 7 Log in to FusionInsight Manager and change the alarm threshold and alarm trigger count based on the actual database connection usage.

Choose **O&M**> **Alarm** > **Thresholds** > *Name of the desired cluster* > **DBService** > **Database** > **Database Connections Usage (DBServer)**. In the **Database Connections Usage (DBServer)** area, click the pencil icon next to **Trigger Count**. In the displayed dialog box, change the trigger count, as shown in **Figure 10-41**.

NOTE

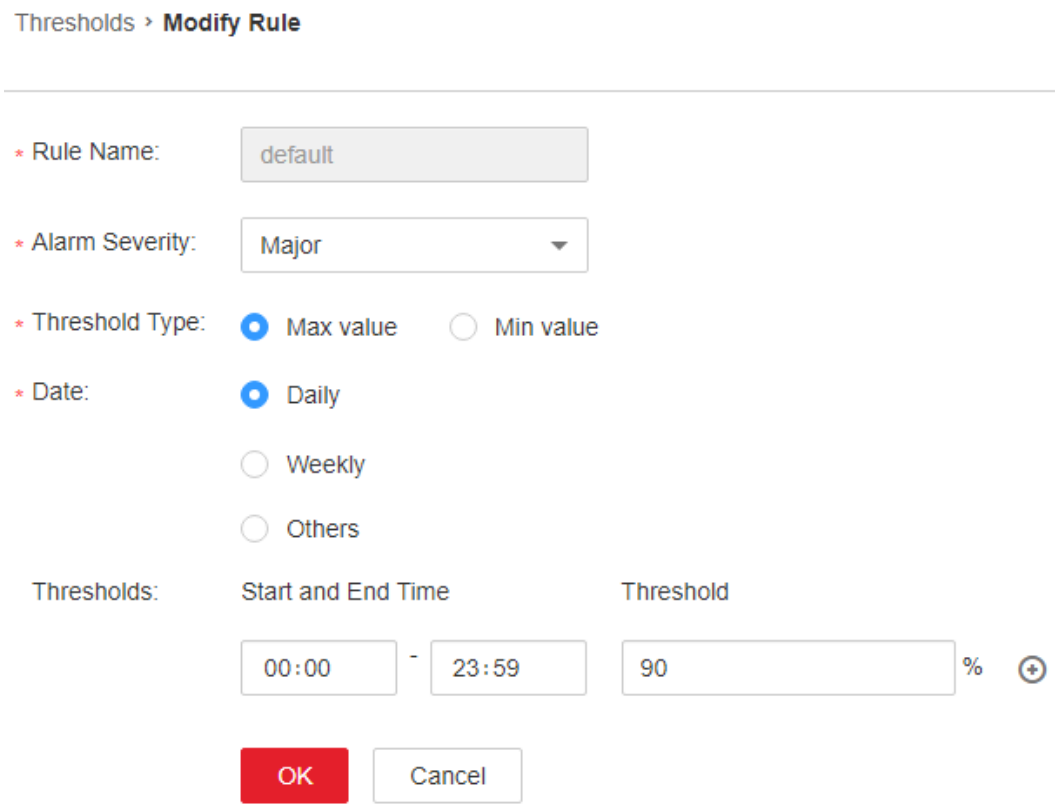
Trigger Count: If the usage of the number of database connections exceeds the threshold consecutively for more than the value of this parameter, an alarm is generated.

Figure 10-41 Setting alarm trigger count



Based on the actual database connection usage, choose **O&M** >**Alarm** > **Thresholds** > *Name of the desired cluster* > **DBService** > **Database** > **Database Connections Usage (DBServer)**. In the **Database Connections Usage (DBServer)** area, click **Modify** in the **Operation** column. In the **Modify Rule** dialog box, modify the required parameters and click **OK** as shown in **Figure 10-42**.

Figure 10-42 Set alarm threshold



Step 8 Wait for 2 minutes and check whether the alarm is automatically cleared.

- If it is, no further action is required.
- If it is not, go to **Step 9**.

Collect fault information

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 Select **DBService** in the required cluster from the **Service**.

Step 11 Specify the host for collecting logs by setting the **Host** parameter that is optional. By default, all hosts are selected.

Step 12 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 13 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.188 ALM-27006 Disk Space Usage of the Data Directory Exceeds the Threshold

Description

The system checks the disk space usage of the data directory on the active DBServer node every 30 seconds and compares the disk usage with the threshold. The alarm is generated when the disk space usage exceeds the threshold for five consecutive times (the default value). The number of consecutive times is configurable. The disk space usage threshold of the data directory is set to 80% by default, which is configurable as well.

The value of **hit number** is configurable. When the value is set to **1** and the disk space usage is lower than or equal to the threshold, the alarm is cleared. When the value is greater than 1 and the disk space usage is lower than 90% of the threshold, the alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
27006	Major	Yes

Parameters

Name	Meaning
ClusterName	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
PartitionName	Specifies the disk partition where the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the actual indicator value exceeds this threshold, the alarm is generated.

Impact on the System

- Service processes become unavailable.
- When the disk space usage of the data directory exceeds 90%, the database reports the "Database Enters the Read-Only Mode" alarm and enters the read-only mode, which may cause service data loss.

Possible Causes

- The alarm threshold is improperly configured.
- The data volume of the database is too large or the disk configuration cannot meet service requirements, causing excessive disk usage.

Procedure

Check whether the threshold is set properly.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **DBService** > **Database** > **Disk Space Usage of the Data Directory** to check whether the alarm threshold is proper (the default value 80% is a proper value).

- If yes, go to **Step 3**.
- If no, go to **Step 2**.

Step 2 Change the alarm threshold based on the actual service situation.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService**. On the **Dashboard** page, view the **Disk Space Usage of the Data Directory** chart and check whether the disk space usage of the data directory is lower than the threshold.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Wait 2 minutes and check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether large files are incorrectly written into the disk.

Step 5 Log in to the active DBService node as user **omm**.

Step 6 Run the following commands to view the files whose size exceeds 500 MB in the data directory and check whether there are large files incorrectly written into the directory:

source $DBSERVER_HOME/.dbservice_profile

find "$DBSERVICE_DATA_DIR"/../ -type f -size +500M

- If yes, go to **Step 7**.
- If no, go to **Step 8**.

Step 7 Handle the large files based on the actual scenario and check whether the alarm is cleared 2 minutes later.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 9 Expand the **Service** drop-down list, and select **DBService** for the target cluster.

Step 10 Specify the host for collecting logs by setting the **Host** parameter which is optional. By default, all hosts are selected.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.189 ALM-27007 Database Enters the Read-Only Mode

Description

The system checks the disk space usage of the data directory on the active DBServer node every 30 seconds. The alarm is generated when the disk space usage exceeds 90%.

The alarm is cleared when the disk space usage is lower than 80%.

Attribute

Alarm ID	Alarm Severity	Auto Clear
27007	Critical	Yes

Parameters

Name	Meaning
ClusterName	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the actual indicator value exceeds this threshold, the alarm is generated.

Impact on the System

The database enters the read-only mode, causing service data loss.

Possible Causes

The disk configuration cannot meet service requirements. The disk usage reaches the upper limit.

Procedure

Check whether the disk space usage reaches the upper limit.

Step 1 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService**.

Step 2 On the **Dashboard** page, view the **Disk Space Usage of the Data Directory** chart and check whether the disk space usage of the data directory exceeds 90%.

- If yes, go to **Step 3**.
- If no, go to **Step 13**.

Step 3 Log in to the active management node of the DBServer as user **omm** and run the following commands to check whether the database enters the read-only mode:

source $DBSERVER_HOME/.dbservice_profile

gsql -U omm -W *password* **-d postgres -p 20051**

show default_transaction_read_only;

NOTE

In the preceding commands, *password* indicates the password of user **omm** of the DBService database. You can run the **\q** command to exit the database.

Check whether the value of **default_transaction_read_only** is **on**.

```
POSTGRES=# show default_transaction_read_only;
 default_transaction_read_only
-------------------------------
 on
(1 row)
```

- If yes, go to **Step 4**.
- If no, go to **Step 13**.

Step 4 Run the following commands to open the **dbservice.properties** file:

source $DBSERVER_HOME/.dbservice_profile

vi ${DBSERVICE_SOFTWARE_DIR}/tools/dbservice.properties

Step 5 Change the value of **gaussdb_readonly_auto** to **OFF**.

Step 6 Run the following command to open the **dbservice.properties** file:

vi ${DBSERVICE_DATA_DIR}/postgresql.conf

Step 7 Delete **default_transaction_read_only = on**.

Step 8 Run the following command for the configuration to take effect:

gs_ctl reload -D ${DBSERVICE_DATA_DIR}

Step 9 Log in to FusionInsight Manager and choose **O&M** > **Alarm** > **Alarms**. On the right of the alarm "Database Enters the Read-Only Mode", click **Clear** in the **Operation** column. In the dialog box that is displayed, click **OK** to manually clear the alarm.

Step 10 Log in to the active management node of the DBServer as user **omm** and run the following commands to view the files whose size exceeds 500 MB in the data directory and check whether there are large files incorrectly written into the directory:

source $DBSERVER_HOME/.dbservice_profile

find "$DBSERVICE_DATA_DIR"/../ -type f -size +500M

- If yes, go to **Step 11**.
- If no, go to **Step 13**.

Step 11 Handle the files that are incorrectly written into the directory based on the actual scenario.

Step 12 Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **DBService**. On the **Dashboard** page, view the **Disk Space Usage of the Data Directory** chart and check whether the disk space usage is lower than 80%.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collect fault information.

Step 13 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 14 Expand the **Service** drop-down list, and select **DBService** for the target cluster.

Step 15 Specify the host for collecting logs by setting the **Host** parameter which is optional. By default, all hosts are selected.

Step 16 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 17 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.190 ALM-29000 Impala Service Unavailable

Description

The alarm module checks the Impala service status every 30 seconds. This alarm is generated if the Impala service is abnormal.

This alarm is cleared after the Impala service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

When the Impala service is abnormal, you cannot perform cluster operations on Impala through FusionInsight Manager. The Impala service functions are unavailable.

Possible Causes

- The Hive service is abnormal.
- The KrbServer service is abnormal.
- The Impala process is abnormal.

Procedure

Check whether the services on which Impala depends are normal.

Step 1 On FusionInsight Manager, choose **Cluster** > **Services** to check whether Hive and KrbServer are stopped.

- If yes, start the stopped services and go to **Step 2**.
- If no, go to **Step 3**.

Step 2 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. In the alarm list, check whether the Impala Service Unavailable alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Step 3 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. In the alarm list, check whether ALM-16004 Hive Service Unavailable and ALM-25500 KrbServer Service Unavailable exist.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Rectify the fault by following the handling procedure of ALM-16004 Hive Service Unavailable or ALM-25500 KrbServer Service Unavailable. Then, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Check whether the Impala process is normal.

Step 5 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. Check whether ALM-12007 Process Fault exists in the alarm list.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 6 Rectify the fault by referring to the handling method of ALM-12007 Process Fault, and then check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect the fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Expand the **Service** drop-down list, and select **Impala** for the target cluster.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.191 ALM-29004 Impalad Process Memory Usage Exceeds the Threshold

Description

The system checks the memory usage of the Impalad process every 30 seconds. This alarm is generated when the system detects that the memory usage exceeds the default threshold (80%).

This alarm is automatically cleared when the system detects that the memory usage of the process falls below the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29004	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

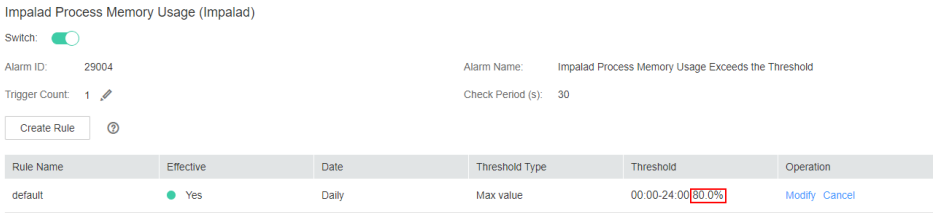
The memory usage is too high. Some query tasks may fail due to insufficient memory.

Possible Causes

The Impalad process is executing a large number of query tasks.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds** > **Impala** > **CPU and Memory** > **Impalad Process Memory Usage (Impalad)** to check the threshold.



Step 2 If the alarm threshold is smaller than 80%, increase the alarm threshold as required and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Step 3 If the threshold is greater than 80%, check whether a large number of concurrent query tasks exist when the alarm is generated. A large number of concurrent query tasks will cause the memory usage to increase sharply. After the tasks are complete, the alarm is automatically cleared. During this period, some tasks may fail to be executed or canceled due to insufficient memory. In this case, try again.

NOTE

If the memory usage always exceeds the threshold, the cluster capacity may need to be expanded.

----End

Alarm Clearing

The alarm is automatically cleared after the burst concurrent tasks are complete.

Related Information

None

10.13.192 ALM-29005 Number of JDBC Connections to Impalad Exceeds the Threshold

Description

The system checks the number of client connections to the Impalad node every 30 seconds. This alarm is generated when the number of client connections exceeds the customized threshold (60 by default).

This alarm is automatically cleared when the number of client connections is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

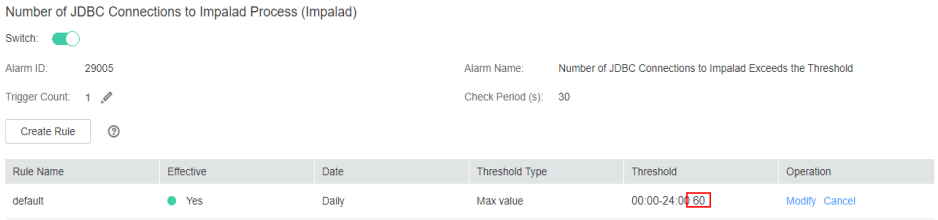
New client connections may be blocked or even fail.

Possible Causes

The number of client connections maintained by the Impalad service is too large or the threshold is too small.

Procedure

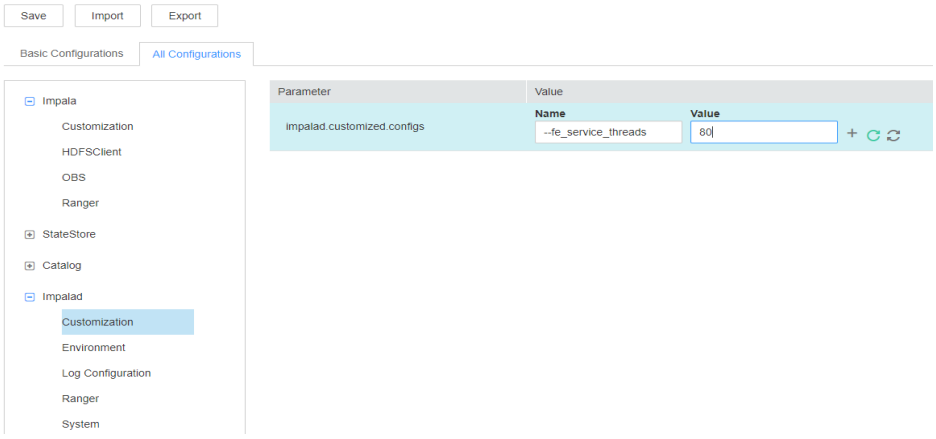
Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds** > **Impala** > **Connections** > **Number of JDBC Connections to Impalad Process** to check the configured threshold.



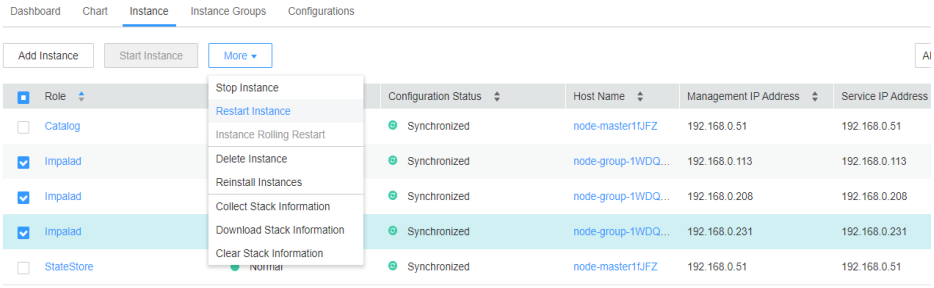
Step 2 Check the number of JDBC applications connected to Impalad and stop idle applications. Check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 3** to change the number of concurrent client connections.

Step 3 On FusionInsight Manager, choose **Cluster** > **Impala** > **Configurations** > **All Configurations** > **Impalad** > **Customization**. Add the customized parameter **--fe_service_threads**. The default value of this parameter is **64**. Change the value as required and click **Save**.



Step 4 After the query tasks on all clients are complete, click the **Instance** tab. Select all Impalad instances, and restart them.



Step 5 After the restart is complete, the alarm is cleared. Run the application that uses JDBC to connect to Impalad again.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.193 ALM-29006 Number of ODBC Connections to Impalad Exceeds the Threshold

Description

The system checks the number of client connections to the Impalad node every 30 seconds. This alarm is generated when the number of client connections exceeds the customized threshold (60 by default).

This alarm is automatically cleared when the number of client connections is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

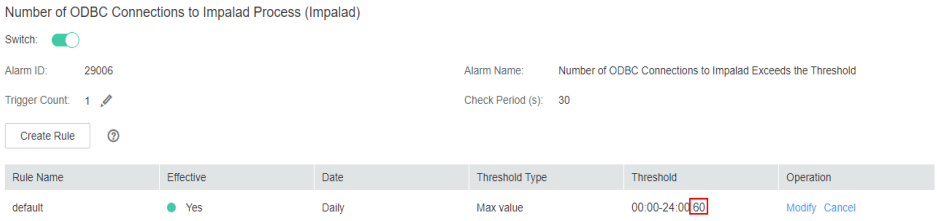
New client connections may be blocked or even fail.

Possible Causes

The number of client connections maintained by the Impalad service is too large or the threshold is too small.

Procedure

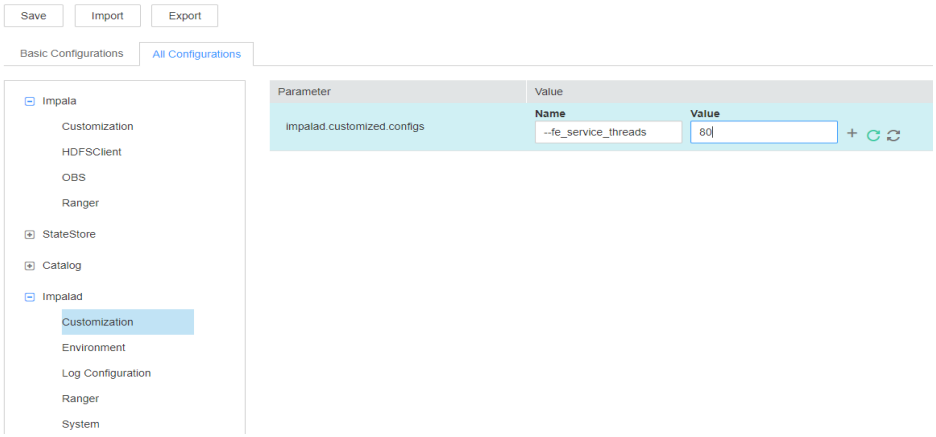
Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Thresholds** > **Impala** > **Connections** > **Number of ODBC Connections to Impalad Process (Impalad)** to check the threshold.



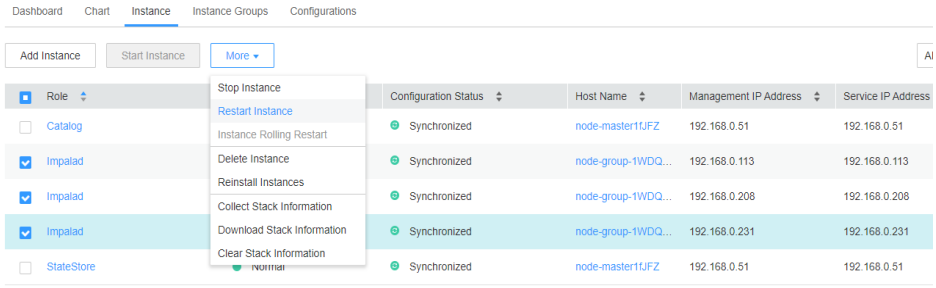
Step 2 Check the number of ODBC applications connected to Impalad and stop idle applications. Check whether the alarm is automatically cleared.

- If yes, no further action is required.
- If no, go to **Step 3** to change the number of concurrent connections supported by Impalad.

Step 3 On FusionInsight Manager, choose **Cluster** > **Impala** > **Configurations** > **All Configurations** > **Impalad** > **Customization**. Add the customized parameter **--fe_service_threads**. The default value of this parameter is **64**. Change the value as required and click **Save**.



Step 4 After the query tasks on all clients are complete, click the **Instance** tab. Select all Impalad instances, and restart them.



Step 5 After the restart is complete, the alarm is cleared. Run the application that uses ODBC to connect to Impalad again.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.194 ALM-29100 Kudu Service Unavailable

Description

The system checks the Kudu service status every 60 seconds. This alarm is generated when the system detects that all Kudu instances are abnormal and considers that the Kudu service is unavailable.

This alarm is cleared when at least one Kudu instance becomes normal and the system considers that the Kudu instance service is restored.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29100	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Users cannot use the Kudu service.

Possible Causes

Some Kudu instances are abnormal.

Procedure

Handle the Kudu instance exceptions.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. On the displayed page, locate the alarm ALM-29100 Kudu Service Unavailable.

Step 2 In the **Location Information** column, record the host name and role name.

Step 3 Choose **Cluster** > **Services** > **Kudu** > **Instance**. Click the role name corresponding to the host name in **Step 2** to restore the instance. Then, check whether the alarm is cleared.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Choose **O&M** > **Alarm** > **Alarms** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 6 In the **Service** area, select the following nodes of the desired cluster.

- Kudu

Step 7 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.195 ALM-29104 Tserver Process Memory Usage Exceeds the Threshold

Description

The system checks the memory usage of the Kudu Tserver process every 60 seconds. This alarm is generated when the system detects that the memory usage of the Kudu Tserver process exceeds the threshold.

This alarm is cleared when the memory usage of the Tserver process becomes normal and the system considers that the Kudu instance service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29104	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Users cannot use the Kudu service.

Possible Causes

The memory usage of a KuduTserver instance is too high.

Procedure

Handle the Kudu instance exceptions.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. On the displayed page, locate the alarm ALM-29104 Tserver Process Memory Usage Exceeds the Threshold.

Step 2 Choose **O&M** > **Alarm** > **Threshold Configuration** > **Kudu**. Locate the alarm threshold and compare it with the memory monitoring item of the cluster Kudu instance to check whether the memory usage exceeds the threshold. If the memory usage exceeds the threshold, rectify the fault or change the threshold.

Step 3 Choose **O&M** > **Alarm** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to step 4.

Collect the fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 In the **Service** area, select the following nodes of the desired cluster.

- Kudu

Step 6 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.196 ALM-29106 Tserver Process CPU Usage Exceeds the Threshold

Description

The system checks the Kudu service status every 60 seconds. This alarm is generated when the system detects that the CPU usage of the Kudu Tserver process is too high.

This alarm is cleared when the CPU usage of the Tserver process becomes normal and the system considers that the Kudu instance service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
29106	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Users cannot use the Kudu service.

Possible Causes

The CPU usage of a KuduTserver instance is too high.

Procedure

Handle the Kudu instance exceptions.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm**. On the displayed page, check whether ALM-29106 Tserver Process CPU Usage Exceeds the Threshold is generated.

- If yes, go to 2.
- If no, go to step 4.

Step 2 Choose **O&M** > **Alarm** > **Thresholds** > **Kudu**. Locate the alarm threshold and check whether the CPU usage of the cluster Kudu instance exceeds the threshold. If yes, rectify the fault or change the threshold.

Step 3 Choose **O&M** > **Alarm** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to step 4.

Collect the fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 In the **Service** area, select the following nodes of the desired cluster.

- Kudu

Step 6 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.197 ALM-29107 Tserver Process Memory Usage Exceeds the Threshold

Description

The system checks the Kudu service status every 60 seconds. This alarm is generated when the memory usage of the Kudu Tserver process exceeds the threshold.

This alarm is cleared when the memory usage of the Tserver process becomes normal and the system considers that the Kudu instance service recovers.

Attribute

Alarm ID	Alarm Severity	Auto Clear
29107	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Users cannot use the Kudu service.

Possible Causes

The memory usage of the KuduTserver instance is too high.

Procedure

Handle the Kudu instance exceptions.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm**. On the displayed page, locate the alarm ALM-29107 Tserver Process Memory Usage Exceeds the Threshold and view the alarm source.

Step 2 Choose **O&M** > **Alarm** > **Threshold Configuration** > **Kudu**, find the threshold of the alarm, compare the memory usage of the KuduTserver instance in the cluster with the threshold, and find the node whose memory usage exceeds the threshold.

Add nodes or reschedule jobs to reduce the memory usage of the Tserver node or change the threshold.

Step 3 Choose **O&M** > **Alarm** and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to step 4.

Collect the fault information.

Step 4 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 5 In the **Service** area, select the following nodes of the desired cluster.

- Kudu

Step 6 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.198 ALM-38000 Kafka Service Unavailable

Description

The system checks the Kafka service status every 30 seconds. This alarm is generated when the Kafka service is unavailable.

This alarm is cleared when the Kafka service recovers.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38000	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The cluster cannot provide the Kafka service, and users cannot perform new Kafka tasks.

Possible Causes

- The KrbServer service is abnormal.(Skip this step if the normal mode is used.)
- The ZooKeeper service is abnormal or does not respond.
- The Broker instance in the Kafka cluster are abnormal.

Procedure

Check the status of the KrbServer service. (Skip this step if the normal mode is used.)

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services > KrbServer**.

Step 2 Check whether the running status of the KrbServer service is **Normal**.

- If yes, go to **Step 5**.
- If no, go to **Step 3**.

Step 3 Rectify the fault by following the steps provided in **ALM-25500 KrbServer Service Unavailable**.

Step 4 Perform **Step 2** again.

Check the status of the ZooKeeper cluster.

Step 5 Check whether the running status of the ZooKeeper service is **Normal**.

- If yes, go to **Step 8**.
- If no, go to **Step 6**.

Step 6 If ZooKeeper service is stopped, start it, else rectify the fault by following the steps provided in **ALM-13000 ZooKeeper Service Unavailable**.

Step 7 Perform **Step 5** again.

Check the Broker status.

Step 8 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance** to go to the Kafka instances page.

Step 9 Check whether all instances in **Roles** are running properly.

- If yes, go to **Step 11**.
- If no, go to **Step 10**.

Step 10 Select all Broker instances, choose **More** > **Restart Instance**, and check whether the instances restart successfully.

- If yes, go to **Step 11**.
- If no, go to **Step 13**.

Step 11 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** to check whether the running status is **Normal**.

- If yes, go to **Step 12**.
- If no, go to **Step 13**.

Step 12 Wait for 30 seconds and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 13**.

Collecting Fault Information

Step 13 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 14 Select **Kafka** in the required cluster from the **Service** drop-down list.

Step 15 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 16 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.199 ALM-38001 Insufficient Kafka Disk Capacity

Description

The system checks the Kafka disk usage every 60 seconds and compares the actual disk usage with the threshold. The disk usage has a default threshold. This alarm is generated when the disk usage is greater than the threshold.

You can change the threshold in **O&M** > **Alarm** > **Thresholds**. Under the service list, choose **Kafka > Disk > Broker Disk Usage (Broker)** and change the threshold.

When the **Trigger Count** is 1, this alarm is cleared when the Kafka disk usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the Kafka disk usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38001	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
PartitionName	Specifies the disk partition where the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Kafka data write operations are affected.

Possible Causes

- The configuration (such as number and size) of the disks for storing Kafka data cannot meet the requirement of the current service traffic, due to which the disk usage reaches the upper limit.
- Data retention time is too long, due to which the data disk usage reaches the upper limit.
- The service plan does not distribute data evenly, due to which the usage of some disks reaches the upper limit.

Procedure

Check the disk configuration of Kafka data.

Step 1 On the FusionInsight Manager portal and click **O&M** > **Alarm** > **Alarms**.

Step 2 In the alarm list, locate the alarm and obtain **HostName** from **Location**.

Step 3 Click **Cluster** > *Name of the desired cluster* > **Hosts**.

Step 4 In the host list, click the host name obtained in **Step 2**.

Step 5 Check whether the **Disk** area contains the partition name in the alarm.

- If yes, go to **Step 6**.
- If no, manually clear the alarm and no further operation is required.

Step 6 Check whether the disk partition usage contained in the alarm reaches 100% in the **Disk** area.

- If yes, handle the alarm by following the instructions in **Related Information**.
- If no, go to **Step 7**.

Check the Kafka data storage duration.

Step 7 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Configurations**.

Step 8 Check whether the value of parameter **disk.adapter.enable** is set to **true**.

- If yes, go to **Step 10**.
- If no, go to **Step 9**.

Step 9 Set the value of **disk.adapter.enable** to **true**. Check whether the value of **adapter.topic.min.retention.hours** is properly set.

- If yes, go to **Step 10**.
- If no, adjust the data retention period based on service requirements.

NOTICE

If the disk auto-adaptation function is enabled, some historical data of specified topics is deleted. If the retention period of some topics cannot be adjusted, click **All Configurations** and add the topics to the value of the **disk.adapter.topic.blacklist** parameter.

Step 10 Wait 10 minutes and check whether the usage of faulty disks reduces.

- If yes, wait until the alarm is cleared.
- If no, go to **Step 11**.

Check the Kafka data plan.

Step 11 In the **Instance** area, click **Broker**. In the **Real Time** area of Broker, Click the drop-down menu in the Chart area and choose **Customize** to customize monitoring items.

Step 12 In the dialog box, select **Disk** > **Broker Disk Usage** and click **OK**.

The Kafka disk usage information is displayed.

Step 13 View the information in **Step 12** to check whether there is only the disk parathion for which the alarm is generated in **Step 2**.

- If yes, go to **Step 14**.
- If no, go to **Step 15**.

Step 14 Perform disk planning and mount a new disk again. Go to the **Instance Configurations** page of the node for which the alarm is generated, modify **log.dirs**, add other disk directories, and restart the Kafka instance.

Step 15 Determine whether to shorten the data retention time configured on Kafka based on service requirements and service traffic.

- If yes, go to **Step 16**.

- If no, go to **Step 17**.

Step 16 Log in to FusionInsight Manager, select **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Configurations**, and click **All Configurations**. In the search box on the right, enter **log.retention.hours**. The value of the parameter indicates the default data retention time of the topic. You can change the value to a smaller one.

NOTE

- For a topic whose data retention time is configured alone, the modification of the data retention time on the Kafka Service Configuration page does not take effect.
- To modify the data retention time for a topic, use the Kafka client command-line interface (CLI) to configure the topic.

  Example: **kafka-topics.sh --zookeeper "***ZooKeeper IP address***:2181/kafka" --alter --topic "***Topic bane***" --config retention.ms= "***retention time***"**

Step 17 Check whether the usage of some disks reaches the upper limit due to unreasonable configuration of the partitions of some topics. For example, the number of partitions configured for a topic with large data volume is smaller than the number of disks. In this case, the data is not evenly allocated to disks.

NOTE

If you do not know which topic has large data volume, you can log in to an instance node based on the host node information obtained in **Step 2**, and go to the data directory (directory specified by **log.dirs** before the modification in **Step 14**) to check whether there is topic with partition that use large disk space.

- If yes, go to **Step 18**.
- If no, go to **Step 19**.

Step 18 In the Kafka client CLI, run the following command to perform partition capacity expansion for the topic:

kafka-topics.sh --zookeeper "*ZooKeeper IP address***:2181/kafka" --alter --topic "***Topic name***" --partitions="***New number of partitions***"**

NOTE

- You are advised to set the new number of partitions to a multiple of the number of Kafka data disks.
- The step may not quickly clear the alarm, and you need to modify the data retention time in **Step 11** to gradually balance data allocation.

Step 19 Determine whether to perform capacity expansion.

NOTE

You are advised to perform capacity expansion for Kafka when the current disk usage exceeds 80%.

- If yes, go to **Step 20**.
- If no, go to **Step 21**.

Step 20 Expand the disk capacity and check whether the alarm is cleared after capacity expansion.

- If yes, no further action is required.
- If no, go to **Step 22**.

Step 21 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 22**.

Collect fault information.

Step 22 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 23 Select **Kafka** in the required cluster from the **Service** drop-down list.

Step 24 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 25 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

Step 1 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**, stop the Broker instance whose status is **Restoring**, record the management IP address of the node where the Broker instance is located, and record **broker.id**. The value can be obtained by using the following method: Click the role name. On the **Configurations** page, select **All Configurations**, and search for the **broker.id** parameter.

Step 2 Log in to the recorded management IP address as user **root**, and run the **df -lh** command to view the mounted directory whose disk usage is 100%, for example, **${BIGDATA_DATA_HOME}/kafka/data1**.

Step 3 Go to the directory, run the **du -sh *** command to view the size of each file in the directory,check whether files other than **kafka-logs** exist, and determine whether these files can be deleted or migrated.

- If yes, go to **Step 8**.
- If no, go to **Step 4**.

Step 4 Go to the **kafka-logs** directory, run the **du -sh *** command, select a partition folder to be moved. The naming rule is **Topic name-Partition ID**. Record the topic and partition.

Step 5 Modify the **recovery-point-offset-checkpoint** and **replication-offset-checkpoint** files in the **kafka-logs** directory in the same way.

1. Decrease the number in the second line in the file. (To remove multiple directories, the number deducted is equal to the number of files to be removed.)
2. Delete the line of the to-be-removed partition. (The line structure is "Topic name Partition ID Offset". Save the data before deletion. Subsequently, the content must be added to the file of the same name in the destination directory.)

Step 6 Modify the **recovery-point-offset-checkpoint** and **replication-offset-checkpoint** files in the destination data directory. For example, **${BIGDATA_DATA_HOME}/kafka/data2/kafka-logs** in the same way.

- Increase the number in the second line in the file. (To move multiple directories, the number added is equal to the number of files to be moved.)
- Add the to-be moved partition to the end of the file. (The line structure is "Topic name Partition ID Offset". You can copy the line data saved in **Step 5**.)

Step 7 Move the partition to the destination directory. After the partition is moved, run the **chown omm:wheel -R** *Partition directory* command to modify the directory owner group for the partition.

Step 8 Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance** to start the Broker instance.

Step 9 Wait for 5 to 10 minutes and check whether the health status of the Broker instance is **Normal**.

- If yes, resolve the disk capacity insufficiency problem according to the handling method of "ALM-38001 Insufficient Kafka Disk Space" after the alarm is cleared.
- If no, contact the O&M personnel.

----End

10.13.200 ALM-38002 Kafka Heap Memory Usage Exceeds the Threshold

Description

The system checks the Kafka service status every 30 seconds. The alarm is generated when the heap memory usage of a Kafka instance exceeds the threshold (95% of the maximum memory) for 10 consecutive times.

When the **Trigger Count** is 1, this alarm is cleared when the heap memory usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the heap memory usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38002	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available Kafka heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The heap memory of the Kafka instance is overused or the heap memory is inappropriately allocated.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **Kafka Heap Memory Usage Exceeds the Threshold** > **Location**. Check the host name of the instance involved in this alarm.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Process** > **Heap Memory Usage of Kafka**, and click **OK**.

Step 3 Check whether the used heap memory of Kafka reaches 95% of the maximum heap memory specified for Kafka.

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Check the heap memory size configured for Kafka.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Configurations** > **All Configurations**> **Broker(Role)** > **Environment**. Increase the value of **KAFKA_HEAP_OPTS** by referring to the Note.

NOTE

- It is recommended that **-Xmx** and **-Xms** be set to the same value.
- You are advised to view **Heap Memory Usage of Kafka** by referring to **Step 2**, and set the value of **KAFKA_HEAP_OPTS** to twice the value of **Heap Memory Used by Kafka.**

Step 5 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 7 Select **Kafka** in the required cluster from the **Service** drop-down list.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.201 ALM-38004 Kafka Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the Kafka service every 30 seconds. This alarm is generated when the direct memory usage of a Kafka instance exceeds the threshold (80% of the maximum memory) for 10 consecutive times.

When the **Trigger Count** is 1, this alarm is cleared when the direct memory usage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the direct memory usage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38004	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available direct memory of the Kafka service is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the Kafka instance is overused or the direct memory is inappropriately allocated.

Procedure

Check the direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **Kafka Direct Memory Usage Exceeds the Threshold** > **Location** to check the host name of the instance for which the alarm is generated.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down menu in the Chart area and choose **Customize** > **Process** > **Kafka Direct Memory Usage**, and click **OK**.

Step 3 Check whether the used direct memory of Kafka reaches 80% of the maximum direct memory specified for Kafka.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Check the direct memory size configured for the Kafka.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Configurations** > **All Configurations** > **Broker(Role)**

> **Environment** to increase the value of **-Xmx** configured in the **KAFKA_HEAP_OPTS** parameter by referring to the Note.

NOTE

- It is recommended that **-Xmx** and **-Xms** be set to the same value.
- You are advised to view **Kafka Direct Memory Usage** by referring to **Step 2**, and set the value of **KAFKA_HEAP_OPTS** to twice the value of **Direct Memory Used by Kafka.**

Step 5 Save the configuration and restart the Kafka service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 8 Select **Kafka** in the required cluster from the **Service** drop-down list.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.202 ALM-38005 GC Duration of the Broker Process Exceeds the Threshold

Description

The system checks the garbage collection (GC) duration of the Broker process every 60 seconds. This alarm is generated when the GC duration exceeds the threshold (12 seconds by default) for 3 consecutive times.

When the **Trigger Count** is 1, this alarm is cleared when the GC duration is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the GC duration is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38005	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

A long GC duration of the Broker process may interrupt the services.

Possible Causes

The Kafka GC duration of the node is too long or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **GC Duration of the Broker Process Exceeds the Threshold** > **Location**. Check the host name of the instance involved in this alarm.

Step 2 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Process** > **Broker GC Duration per Minute**, and click **OK**.

Step 3 Check whether the GC duration of the Broker process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Check the direct memory size configured for the Kafka.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Configurations** > **All Configurations** > **Broker(Role)** > **Environment** to increase the value of **-Xmx** configured in the **KAFKA_HEAP_OPTS** parameter by referring to the Note.

NOTE

- It is recommended that **-Xmx** and **-Xms** be set to the same value.
- You are advised to set the value of **KAFKA_HEAP_OPTS** to twice the value of **Direct Memory Used by Kafka.**

  On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. Click the instance for which the alarm is generated to go to the page for the instance. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **Process** > **Kafka Direct Memory Resource Status** to check the value of **Direct Memory Used by Kafka**.

Step 5 Save the configuration and restart the Kafka service.

Step 6 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 8 Select **Kafka** in the required cluster from the **Service** drop-down list.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.203 ALM-38006 Percentage of Kafka Partitions That Are Not Completely Synchronized Exceeds the Threshold

Description

The system checks the percentage of Kafka partitions that are not completely synchronized to the total number of partitions every 60 seconds. This alarm is

generated when the percentage exceeds the threshold (50% by default) for 3 consecutive times.

When the **Trigger Count** is 1, this alarm is cleared when the percentage is less than or equal to the threshold. When the **Trigger Count** is greater than 1, this alarm is cleared when the percentage is less than or equal to 90% of the threshold.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

Too many Kafka partitions that are not completely synchronized affect service reliability. In addition, data may be lost when leaders are switched.

Possible Causes

Some nodes where the Broker instance resides are abnormal or stop running. As a result, replicas of some partitions in Kafka are out of the in-sync replicas (ISR) set.

Procedure

Check Broker instances.

Step 1 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. The Kafka instances page is displayed.

Step 2 Check whether faulty nodes exist among all Broker nodes.

- If yes, record the host name of the node and go to **Step 3**.

- If no, go to **Step 5**.

Step 3 On the FusionInsight Manager portal, click **O&M** > **Alarm** > **Alarms** to check whether the fault described in **Step 2** exists in the alarm information and handle the alarm based on corresponding methods.

Step 4 On the FusionInsight Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. The Kafka instances page is displayed.

Step 5 Check whether stopped nodes exist among all Broker instance.

- If yes, go to **Step 6**.
- If no, go to **Step 7**.

Step 6 Select all stopped Broker instances and click **Start Instance**.

Step 7 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On the FusionInsight Manager portal, choose **O&M** > **Log** > **Download**.

Step 9 Select **Kafka** in the required cluster from the **Service** drop-down list.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.204 ALM-38007 Status of Kafka Default User Is Abnormal

Description

The system checks the default user of Kafka every 60 seconds. This alarm is generated when the system detects that the user status is abnormal.

Trigger Count is set to **1**. This alarm is cleared when the user status becomes normal.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38007	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host name for which the alarm is generated.
Trigger Condition	Specifies the condition that the Kafka default user status is abnormal.

Impact on the System

If the Kafka default user status is abnormal, metadata synchronization between Brokers and interaction between Kafka and ZooKeeper will be affected, affecting service production, consumption, and topic creation and deletion.

Possible Causes

- The Sssd service is abnormal.
- Some Broker instances stop running.

Procedure

Check whether the Sssd service is abnormal.

Step 1 On the FusionInsight Manager portal, choose **O&M** > **Alarm** > **Alarms** > **Status of Kafka Default User Is Abnormal** > **Location** to check the host name of the instance for which the alarm is generated.

Step 2 Find the host information in the alarm information and log in to the host.

Step 3 Run the **id -Gn kafka** command and check whether "No such user" is displayed in the command output.

- If yes, record the host name of the node and go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Alarms**. Check whether there is **Sssd Service Exception** in the alarm information. If there is, handle the alarm based on alarm information.

Check the running status of the Broker instance.

Step 5 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. The Kafka instance page is displayed.

Step 6 Check whether there are stopped nodes on all Broker instances.

- If yes, go to **Step 7**.
- If no, go to **Step 8**.

Step 7 Select all stopped Broker instances and click **Start Instance**.

Step 8 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect fault information.

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 In the **Service** area, select **Kafka** in the required cluster.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.205 ALM-38008 Abnormal Kafka Data Directory Status

Description

The system checks the Kafka data directory status every 60 seconds. This alarm is generated when the system detects that the status of a data directory is abnormal.

Trigger Count is set to **1**. This alarm is cleared when the data directory status becomes normal.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host name for which the alarm is generated.
DirName	Specifies the directory name for which the alarm is generated.
Trigger Condition	Specifies the condition that the Kafka data directory status is abnormal.

Impact on the System

If the Kafka data directory status is abnormal, the current replicas of all partitions in the data directory are brought offline, and the data directory status of multiple nodes is abnormal at the same time. As a result, some partitions may become unavailable.

Possible Causes

- The data directory permission is tampered with.
- The disk where the data directory is located is faulty.

Procedure

Check the permission on the faulty data directory.

Step 1 Find the host information in the alarm information and log in to the host.

Step 2 In the alarm information, check whether the data directory and its subdirectories belong to the omm:wheel group.

- If yes, record the host name of the node and go to **Step 4**.
- If no, go to **Step 3**.

Step 3 Restore the owner group of the data directory and its subdirectories to omm:wheel.

- If yes, go to **Step 6**.
- If no, go to **Step 5**.

Check whether the disk where the data directory is located is faulty.

Step 4 In the upper-level directory of the data directory, create and delete files as user **omm**. Check whether data read/write on the disk is normal.

Step 5 Replace or repair the disk where the data directory is located to ensure that data read/write on the disk is normal.

Step 6 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**. On the Kafka instance page that is displayed, restart the Broker instance on the host recorded in **Step 2**.

Step 7 After Broker is started, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 8**.

Collect fault information.

Step 8 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 9 In the **Service** area, select **Kafka** in the required cluster.

Step 10 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 11 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.206 ALM-38009 Busy Broker Disk I/Os

Description

The system checks the I/O status of each Kafka disk every 60 seconds. This alarm is generated when the I/O status of a Kafka data directory disk on a broker exceeds the threshold (80% by default).

The alarm smoothing time is 3. This alarm is cleared when the disk I/O is lower than the threshold (80% by default).

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38009	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Data directory name	Name of the data directory of the Kafka disk with busy I/Os

Impact on the System

The I/O usage of the disk partition is high. Data may fail to be written to the Kafka topic for which the alarm is generated.

Possible Causes

- There are many replicas configured for a topic.
- The parameter specifying producer message batch write is inappropriately configured. The service traffic of this topic is too heavy, and the current partition configuration is inappropriate.

Procedure

Check the number of replication.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. On the displayed page, select this alarm, and check the **TopicName** for which this alarm is generated.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **KafkaTopic Monitor**. Search the topic for which this alarm is generated. On the displayed page, view the number of **replication**.

Step 3 If the number of replication is greater than 3, decrease the value to **3**.

Specifically, run the following command to re-plan replicas of the Kafka topic.

kafka-reassign-partitions.sh --zookeeper *{zk_host}:{port}*/**kafka --reassignment-json-file** *{manual assignment json file path}* **--execute**

For example:

/opt/Bigdata/client/Kafka/kafka/bin/kafka-reassign-partitions.sh --zookeeper 10.149.0.90:2181,10.149.0.91:2181,10.149.0.92:2181/kafka --reassignment-json-file expand-cluster-reassignment.json --execute

NOTE

In the **expand-cluster-reassignment.json** file, describe the Brokers to which the partitions of the topic are migrated in the format of {"partitions":[{"topic": "*topicName*","partition": 1,"replicas": [1,2,3] }],"version":1}.

Step 4 After a period of time, check whether this alarm is cleared. If this alarm persists, go to **Step 5**.

Check the partition planning of the topic.

Step 5 On the **KafkaTopic Monitor** page, click **Topic Traffic** > **Topic Input Traffic** of each topic to obtain the topic with the largest value of **Topic Input Traffic,** and check partitions on this topic and information about hosts of these partitions.

Step 6 Log in to the hosts queried in **Step 5** and run the **iostat -d -x** command to check the value of **%util** for each disk:

```
189-39-172-162:/opt/R3/FusionInsight_Manager/software/packs # iostat -d -x
Linux 3.0.76-0.11-default (189-39-172-162)      06/26/19        _x86_64_

Device:         rrqm/s   wrqm/s     r/s     w/s    rsec/s    wsec/s avgrq-sz avgqu-sz   await  svctm  %util
xvda              0.04    44.44    1.26   21.94     43.62    531.02    24.78     0.03    1.44   0.56   1.30
xvde              0.16   431.84   13.78   82.51    284.32   4115.90    45.70     0.06    1.41   0.64   6.21
```

- If the value is high for each disk, expand the Kafka disks. After the capacity expansion, plan partitions of the topic by following the instruction in **Step 3**.
- If values of **%util** for the disks vary greatly, check the disk partition configuration of Kafka.For example: The configuration item indicates **log.dirs** in the **server.properties** file in the **${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/1_14_Broker/etc** directory.

  Run the following command to view information about the Filesystem:

  df -h log.dirs *configuration item*.

  The command output is as follows:

  ```
  189-39-172-162:/opt/R3/FusionInsight_Manager/software/packs # df -h /srv/BigData/kafka/data1/kafka-logs/
  Filesystem      Size  Used Avail Use% Mounted on
  /dev/xvda2       36G   21G   14G  62% /
  189-39-172-162:/opt/R3/FusionInsight_Manager/software/packs #
  ```

- If the partition of the Filesystem matches the partition with the high **%util**, plan Kafka partitions on idle disks, and set **log.dirs** to directories of the idle disk. Then, plan partitions of the topic by following the instruction in **Step 3**. to ensure that the partitions of the topic are evenly distributed to disks.

Step 7 After a period of time, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, repeat **Step 5** to **Step 6** for three times. If the number of repeated execution times reaches the upper limit, go to **Step 8**.

Step 8 After a period of time, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 9**.

Collect fault information.

Step 9 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 10 In the **Service** area, select **Kafka** in the required cluster.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.207 ALM-38010 Topics with Single Replica

Description

The system checks the number of replicas of each topic every 60 seconds on the node where the Kafka Controller resides. This alarm is generated when there is one replica for a topic.

Attribute

Alarm ID	Alarm Severity	Automatically Cleared
38010	Warning	No

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
TopicName	Specifies the list of topics for which the alarm is generated.

Impact on the System

There is the single point of failure (SPOF) risk for topics with only one replica. When the node where the replica resides becomes abnormal, the partition does not have a leader, and services on the topic are affected.

Possible Causes

- The number of replicas for the topic is incorrectly configured.

Procedure

Check the number of replicas for the topic.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**, click ⌄ of this alarm, and view the **TopicName** list in **Location**.

Step 2 Check whether replicas need to be added for the topic for which the alarm is generated.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 On the FusionInsight client, re-plan topic replicas and describe the partition distribution of the topic in the **add-replicas-reassignment.json** file in the following format: {"partitions":[{"topic": "*topic name*","partition": 1,"replicas": [1,2] }],"version":1}. Then, run the following command to add replicas:

kafka-reassign-partitions.sh --zookeeper *{zk_host}:{port}*/**kafka --reassignment-json-file** *{manual assignment json file path}* **--execute**

For example:

/opt/Bigdata/client/Kafka/kafka/bin/kafka-reassign-partitions.sh --zookeeper 192.168.0.90:2181,192.168.0.91:2181,192.168.0.92:2181/kafka --reassignment-json-file add-replicas-reassignment.json --execute

Step 4 Run the following command to check the task execution progress:

kafka-reassign-partitions.sh --zookeeper *{zk_host}:{port}*/**kafka --reassignment-json-file** *{manual assignment json file path}* **--verify**

For example:

/opt/Bigdata/client/Kafka/kafka/bin/kafka-reassign-partitions.sh --zookeeper 192.168.0.90:2181,192.168.0.91:2181,192.168.0.92:2181/kafka --reassignment-json-file add-replicas-reassignment.json --verify

Step 5 After completing the handling operations or confirming that the alarm has no impact, manually clear the alarm on FusionInsight Manager.

Step 6 After a period of time, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 In the **Service** area, select **Kafka** in the required cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

If the alarm has no impact, manually clear the alarm.

Related Information

None

10.13.208 ALM-43001 Spark2x Service Unavailable

Description

The system checks the Spark2x service status every 300 seconds. This alarm is generated when the Spark2x service is unavailable.

This alarm is cleared when the Spark2x service recovers.

Attribute

Alarm ID	Alarm Severity	Auto Clear
43001	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The tasks submitted by users fail to be executed.

Possible Causes

- The KrbServer service is abnormal.
- The LdapServer service is abnormal.
- The ZooKeeper service is abnormal.
- The HDFS service is abnormal.
- The Yarn service is abnormal.
- The corresponding Hive service is abnormal.
- Spark2x assembly packet is abnormal.

Procedure

If the alarm is abnormal Spark2x assembly packet, the Spark packet is abnormal. Wait for about 10 minutes. The alarm is automatically cleared.

Check whether service unavailability alarms exist in services depended by **Spark2x.**

Step 1 On FusionInsight Manager, choose **O&M > Alarm > Alarms**.

Step 2 Check whether the following alarms exist in the alarm list:

- ALM-25500 KrbServer Service Unavailable
- ALM-25000 LdapServer Service Unavailable
- ALM-13000 ZooKeeper Service Unavailable
- ALM-14000 HDFS Service Unavailable
- ALM-18000 Yarn Service Unavailable
- ALM-16004 Hive Service Unavailable

NOTE

If the multi-instance function is enabled in the cluster and multiple Spark2x service instances are installed, you need to determine the Spark2x service instance where the alarm is generated based on the value of ServiceName in **Location**. Then you need to check whether the corresponding Hive service is faulty. Spark2x corresponds to Hive, and Spark2x1 corresponds to Hive1. The others follow the same rule.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.

Step 3 Handle the service unavailability alarms based on the troubleshooting methods provided in the alarm help.

After all the service unavailability alarms are cleared, wait a few minutes and check whether this alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Collect fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Select the following nodes in the required cluster from the **Service** (Hive is the specific Hive service determined based on **ServiceName** in the alarm location information).

- KrbServer
- LdapServer
- ZooKeeper
- HDFS
- Yarn
- Hive

Step 6 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.209 ALM-43006 Heap Memory Usage of the JobHistory2x Process Exceeds the Threshold

Description

The system checks the JobHistory2x Process status every 30 seconds. The alarm is generated when the heap memory usage of a JobHistory2x Process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Alarm Severity	Auto Clear
43006	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available JobHistory2x Process heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The heap memory of the JobHistory2x Process is overused or the heap memory is inappropriately allocated.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43006**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JobHistory2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **Memory > JobHistory2x Memory Usage Statistics** from the drop-down list box in the upper right corner and click **OK**. Check whether the used heap memory of the JobHistory2x Process reaches the

threshold(default value is 95%) of the maximum heap memory specified for JobHistory2x.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click **JobHistory2x** by which the alarm is reported to go to the **Dashboard** page, click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Memory > Statistics for the heap memory of the JobHistory2x Process**, and click **OK**. Based on the alarm generation time, check the values of the used heap memory of the JobHistory2x process in the corresponding period and obtain the maximum value.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JobHistory2x** > **Default**. The default value of **SPARK_DAEMON_MEMORY** is 4GB. You can change the value according to the following rules: Ratio of the maximum heap memory usage of the JobHistory2x to the **Threshold** of the **JobHistory2x Heap Memory Usage Statistics (JobHistory2x)** in the alarm period. If this alarm is generated occasionally after the parameter value is adjusted, increase the value by 0.5 times. If the alarm is frequently reported after the parameter value is adjusted, increase the value by 1 time.

NOTE

On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* **> Spark2x** > **Memory** >**JobHistory2x Heap Memory Usage Statistics (JobHistory2x)** to view **Threshold**.

Step 5 Restart all JobHistory2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Spark2x** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.210 ALM-43007 Non-Heap Memory Usage of the JobHistory2x Process Exceeds the Threshold

Description

The system checks the JobHistory2x Process status every 30 seconds. The alarm is generated when the non-heap memory usage of a JobHistory2x Process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Alarm Severity	Auto Clear
43007	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available JobHistory2x Process non-heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The non-heap memory of the JobHistory2x Process is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43007**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JobHistory2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **Memory** > **JobHistory2x Memory Usage Statistics** from the drop-down list box in the upper right corner and click **OK**, Check whether the used non-heap memory of the JobHistory2x Process reaches the threshold(default value is 95%) of the maximum non-heap memory specified for JobHistory2x.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click **JobHistory2x** by which the alarm is reported to go to the **Dashboard** page, click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Memory > Statistics for the non-heap memory of the JobHistory2x Process**, and click **OK**. Based on the alarm generation time, check the values of the used non-heap memory of the JobHistory2x process in the corresponding period and obtain the maximum value.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JobHistory2x** > **Default**. You can change the value of **-XX:MaxMetaspaceSize** in **SPARK_DAEMON_JAVA_OPTS** according to the following rules: Ratio of the JobHistory2x non-heap memory usage to the **Threshold** of **JobHistory2x Non-Heap Memory Usage Statistics (JobHistory2x)** in the alarm period.

NOTE

On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* **> Spark2x** > **Memory** >**JobHistory2x Non-Heap Memory Usage Statistics (JobHistory2x)** to view **Threshold**.

Step 5 Restart all JobHistory2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Spark2x** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.211 ALM-43008 The Direct Memory Usage of the JobHistory2x Process Exceeds the Threshold

Description

The system checks the JobHistory2x Process status every 30 seconds. The alarm is generated when the direct memory usage of a JobHistory2x Process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Alarm Severity	Auto Clear
43008	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available JobHistory2x Process direct memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct memory of the JobHistory2x Process is overused or the direct memory is inappropriately allocated.

Procedure

Check direct memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43008**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JobHistory2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **Memory** > **JobHistory2x Memory Usage Statistics** from the drop-down list box in the upper right corner and click **OK**. Check whether the used direct memory of the JobHistory2x Process reaches the threshold(default value is 95%) of the maximum direct memory specified for JobHistory2x.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click **JobHistory2x** by which the alarm is reported to go to the **Dashboard** page, click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Memory > Direct Memory of JobHistory2x**, and click **OK**. Based on the alarm generation time, check the values of the used direc memory of the JobHistory2x process in the corresponding period and obtain the maximum value.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** ***N****ame of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JobHistory2x** > **Default**. The default value of **-XX:MaxDirectMemorySize** in **SPARK_DAEMON_JAVA_OPTS** is 512 MB. You can change the value according to the following rules: Ratio of the maximum direct memory usage of the JobHistory2x to the **Threshold** of the **JobHistory2x Direct Memory Usage Statistics (JobHistory2x)** in the alarm period. If this alarm is generated occasionally after the parameter value is adjusted, increase the value by 0.5 times. If the alarm is frequently reported after the parameter value is adjusted, increase the value by 1 time. It is recommended that the value be less than or equal to the value of **SPARK_DAEMON_MEMORY**.

NOTE

On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* **> Spark2x** > **Memory** >**JobHistory2x Direct Memory Usage Statistics (JobHistory2x)** to view **Threshold**.

Step 5 Restart all JobHistory2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Spark2x** in the required cluster from the **Service**.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.212 ALM-43009 JobHistory2x Process GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) time of the JobHistory2x Process every 60 seconds. This alarm is generated when the detected GC time exceeds the threshold (exceeds 5 seconds for three consecutive checks.) To change the threshold, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **GC Time** > **Total GC time in milliseconds (JobHistory2x)**. This alarm is cleared when the JobHistory2x GC time is shorter than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
43009	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.

Name	Meaning
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the GC time exceeds the threshold, JobHistory2x maybe run in low performance.

Possible Causes

The memory of JobHistory2x is overused, the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC time.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43009**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JobHistory2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **GC Time** > **Garbage Collection (GC) Time of JobHistory2x** from the drop-down list box in the upper right corner and click **OK** to check whether the GC time is longer than the threshold(default value: 12 seconds).

- If yes, go to **Step 3**.
- If no, go to **Step 6**.

Step 3 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JobHistory2x** > **Default**. The default value of **SPARK_DAEMON_MEMORY** is 4GB. You can change the value according to the following rules: If this alarm is generated occasionally, increase the value by 0.5 times. If the alarm is frequently reported, increase the value by 1 time.

Step 4 Restart all JobHistory2x instances.

Step 5 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager interface of active and standby clusters, choose **O&M** > **Log > Download**.

Step 7 Select **Spark2x** in the required cluster from the **Service**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.213 ALM-43010 Heap Memory Usage of the JDBCServer2x Process Exceeds the Threshold

Description

The system checks the JDBCServer2x Process status every 30 seconds. The alarm is generated when the heap memory usage of a JDBCServer2x Process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Alarm Severity	Auto Clear
43010	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.

Name	Meaning
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available JDBCServer2x Process heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The heap memory of the JDBCServer2x Process is overused or the heap memory is inappropriately allocated.

Procedure

Check heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43010**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JDBCServer2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **Memory** > **JDBCServer2x Memory Usage Statistics** from the drop-down list box in the upper right corner and click **OK**. Check whether the used heap memory of the JDBCServer2x Process reaches the threshold(default value is 95%) of the maximum heap memory specified for JDBCServer2x.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click **JDBCServer2x** by which the alarm is reported to go to the **Dashboard** page, click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Memory > Statistics for the heap memory of the JDBCServer2x Process**, and click **OK**. Based on the alarm generation time, check the values of the used heap memory of the JDBCServer2x process in the corresponding period and obtain the maximum value.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JDBCServer2x** > **Tuning**. The default value of **SPARK_DRIVER_MEMORY** is 4 GB. You can change the value according to the following rules: Ratio of the maximum heap memory usage of the JobHistory2x to the **Threshold** of the

JDBCServer2x Heap Memory Usage Statistics (JDBCServer2x) in the alarm period. If this alarm is generated occasionally after the parameter value is adjusted, increase the value by 0.5 times. If the alarm is frequently reported after the parameter value is adjusted, increase the value by 1 time. In the case of large service volume and high concurrency, add instances.

NOTE

On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* **> Spark2x** > **Memory** >**JDBCServer2x Heap Memory Usage Statistics (JDBCServer2x)** to view **Threshold**.

Step 5 Restart all JDBCServer2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Spark2x** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.214 ALM-43011 Non-Heap Memory Usage of the JDBCServer2x Process Exceeds the Threshold

Description

The system checks the JDBCServer2x Process status every 30 seconds. The alarm is generated when the non-heap memory usage of an JDBCServer2x Process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Alarm Severity	Auto Clear
43011	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available JDBCServer2x Process non-heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The non-heap memory of the JDBCServer2x Process is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43011**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JDBCServer2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **Memory** > **JDBCServer2x Memory Usage Statistics** from the drop-down list box in the upper right corner and click **OK**, Check whether the used non-heap memory of the JDBCServer2x Process reaches the threshold(default value is 95%) of the maximum non-heap memory specified for JDBCServer2x.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click **JDBCServer2x** by which the alarm is reported to go to the **Dashboard** page, click the drop-down list in the upper

right corner of the chart area, choose **Customize** > **Memory > Statistics for the non-heap memory of the JDBCServer2x Process**, and click **OK**. Based on the alarm generation time, check the values of the used non-heap memory of the JDBCServer2x process in the corresponding period and obtain the maximum value.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JDBCServer2x** > **Tuning**. You can change the value of **-XX: MaxMetaspaceSize** in **spark.driver.extraJavaOptions** according to the following rules: Ratio of the JDBCServer2x non-heap memory usage to the **Threshold** of **JDBCServer2x Non-Heap Memory Usage Statistics (JDBCServer2x)** in the alarm period.

NOTE

On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* **> Spark2x** > **Memory** > **JDBCServer2x Non-Heap Memory Usage Statistics (JDBCServer2x)** to view **Threshold**.

Step 5 Restart all JDBCServer2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log >Download**.

Step 8 Select **Spark2x** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.215 ALM-43012 Direct Heap Memory Usage of the JDBCServer2x Process Exceeds the Threshold

Description

The system checks the JDBCServer2x Process status every 30 seconds. The alarm is generated when the direct heap memory usage of a JDBCServer2x Process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Alarm Severity	Auto Clear
43012	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Specifies the threshold triggering the alarm. If the current indicator value exceeds this threshold, the alarm is generated.

Impact on the System

If the available JDBCServer2x Process direct heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct heap memory of the JDBCServer2x Process is overused or the direct heap memory is inappropriately allocated.

Procedure

Check direct heap memory usage.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43012**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JDBCServer2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > Memory > **JDBCServer2x Memory Usage Statistics** from the drop-down list box in the upper right corner and click **OK**. Check whether the used direct heap memory of the JDBCServer2x Process reaches

the threshold(default value is 95%) of the maximum direct heap memory specified for JDBCServer2x.

- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 On the FusionInsight Manager home page, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click **JDBCServer2x** by which the alarm is reported to go to the **Dashboard** page, click the drop-down list in the upper right corner of the chart area, choose **Customize** > **Memory > Direct Memory of JDBCServer2x**, and click **OK**. Based on the alarm generation time, check the values of the used direct memory of the JDBCServer2x process in the corresponding period and obtain the maximum value.

Step 4 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JDBCServer2x** > **Tuning**. The default value of **-XX:MaxDirectMemorySize** in **spark.driver.extraJavaOptions** is 512 MB. You can change the value according to the following rules: Ratio of the maximum direct memory usage of the JDBCServer2x to the **Threshold** of the **JDBCServer2x Direct Memory Usage Statistics (JDBCServer2x)** in the alarm period. If this alarm is generated occasionally after the parameter value is adjusted, increase the value by 0.5 times. If the alarm is frequently reported after the parameter value is adjusted, increase the value by 1 time. In the case of large service volume and high service concurrency, you are advised to add instances.

NOTE

On the FusionInsight Manager home page, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* **> Spark2x** > **Memory** >**JDBCServer2x Direct Memory Usage Statistics (JDBCServer2x)** to view **Threshold**.

Step 5 Restart all JDBCServer2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 7**.

Collect fault information.

Step 7 On the FusionInsight Manager portal, choose **O&M** > **Log > Download**.

Step 8 Select **Spark2x** in the required cluster from the **Service**.

Step 9 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.216 ALM-43013 JDBCServer2x Process GC Time Exceeds the Threshold

Description

The system checks the garbage collection (GC) time of the JDBCServer2x Process every 60 seconds. This alarm is generated when the detected GC time exceeds the threshold (exceeds 5 seconds for three consecutive checks.) To change the threshold, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **GC Time** > **Total GC time in milliseconds (JDBCServer2x)**. This alarm is cleared when the JDBCServer2x GC time is shorter than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
43013	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service name for which the alarm is generated.
RoleName	Specifies the role name for which the alarm is generated.
HostName	Specifies the object (host ID) for which the alarm is generated.
Trigger Condition	Generates an alarm when the actual indicator value exceeds the specified threshold.

Impact on the System

If the GC time exceeds the threshold, JDBCServer2x maybe run in low performance.

Possible Causes

The memory of JDBCServer2x is overused, the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC time.

Step 1 On the FusionInsight Manager portal, choose **O&M > Alarm > Alarms** and select the alarm whose **ID** is **43013**. Check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Instance** and click the JDBCServer2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **GC Time** > **Garbage Collection (GC) Time of JDBCServer2x** from the drop-down list box in the upper right corner and click **OK** to check whether the GC time is longer than the threshold(default value: 12 seconds).

- If yes, go to **Step 3**.
- If no, go to **Step 6**.

Step 3 On the FusionInsight Manager portal, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, and click **All Configurations**. Choose **JDBCServer2x** > **Default**. The default value of **SPARK_DRIVER_MEMORY** is 4 GB. You can change the value according to the following rules: If this alarm is generated occasionally, increase the value by 0.5 times. If the alarm is frequently reported, increase the value by 1 time. In the case of large service volume and high service concurrency, you are advised to add instances.

Step 4 Restart all JDBCServer2x instances.

Step 5 After 10 minutes, check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect fault information.

Step 6 On the FusionInsight Manager interface of active and standby clusters, choose **O&M** > **Log > Download**.

Step 7 Select **Spark2x** in the required cluster from the **Service**.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Related Information

None

10.13.217 ALM-43017 JDBCServer2x Process Full GC Number Exceeds the Threshold

Description

The system checks the number of Full garbage collection (GC) times of the JDBCServer2x process every 60 seconds. This alarm is generated when the detected Full GC number exceeds the threshold (exceeds 12 for three consecutive checks.) You can change the threshold by choosing **O&M > Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **GC number** > **Full GC Number of JDBCServer2x**. This alarm is cleared when the Full GC number of the JDBCServer2x process is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
43017	Major	Yes

Parameters

Name	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

The performance of the JDBCServer2x process is affected, or even the JDBCServer2x process is unavailable.

Possible Causes

The heap memory usage of the JDBCServer2x process is excessively large, or the heap memory is inappropriately allocated. As a result, Full GC occurs frequently.

Procedure

Check the number of Full GCs.

Step 1 Log in to FusionInsight Manager, choose **O&M** > **Alarm > Alarms**, select this alarm, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. On the displayed page, click the JDBCServer2x for which the alarm is reported. On the **Dashboard** page that is displayed, click the drop-down menu in the Chart area and choose **Customize** > **GC Number** > **Full GC Number of JDBCServer2x** in the upper right corner and click **OK**. Check whether the number of Full GCs of the JDBCServer2x process is greater than the threshold(default value: 12).

- If it is, go to **Step 3**.
- If it is not, go to **Step 6**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configurations**. On the displayed page, choose **JDBCServer2x** > **Tuning**. The default value of **SPARK_DRIVER_MEMORY** is 4GB. You can change the value according to the following rules: If this alarm is generated occasionally, increase the value by 0.5 times. If the alarm is frequently reported, increase the value by 1 time. In the case of large service volume and high concurrency, add instances.

Step 4 Restart all JDBCServer2x instances.

Step 5 After 10 minutes, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 Log in to FusionInsight Manager, and choose **O&M** > **Log** > **Download**.

Step 7 Select **Spark2x** in the required cluster from the **Service** drop-down list.

Step 8 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.218 ALM-43018 JobHistory2x Process Full GC Number Exceeds the Threshold

Description

The system checks the number of Full garbage collection (GC) times of the JobHistory2x process every 60 seconds. This alarm is generated when the detected Full GC number exceeds the threshold (exceeds 12 for three consecutive checks.) You can change the threshold by choosing **O&M** > **Alarm > Thresholds** > *Name of the desired cluster* > **Spark2x** > **GC number** > **Full GC Number of JobHistory2x**. This alarm is cleared when the Full GC number of the JobHistory2x process is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
43018	Major	Yes

Parameters

Name	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

The performance of the JobHistory2x process is affected, or even the JobHistory2x process is unavailable.

Possible Causes

The heap memory usage of the JobHistory2x process is excessively large, or the heap memory is inappropriately allocated. As a result, Full GC occurs frequently.

Procedure

Check the number of Full GCs.

Step 1 Log in to FusionInsight Manager, choose **O&M** > **Alarm > Alarms**, select this alarm, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. On the displayed page, click the JobHistory2x for which the alarm is reported. On the **Dashboard** page that is displayed, click the drop-down menu in the Chart area and choose **Customize** > **GC Number** > **Full GC Number of JobHistory2x** in the upper right corner and click **OK**. Check whether the number of Full GCs of the JobHistory2x process is greater than the threshold(default value: 12).

- If it is, go to **Step 3**.
- If it is not, go to **Step 6**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configurations**. On the displayed page, choose **JobHistory2x** > **Default.** The default value of **SPARK_DAEMON_MEMORY** is 4GB. You can change the value according to the following rules: If this alarm is generated occasionally, increase the value by 0.5 times. If the alarm is frequently reported, increase the value by 1 time.

Step 4 Restart all JobHistory2x instances.

Step 5 After 10 minutes, check whether the alarm is cleared.

- If it is, no further action is required.
- If it is not, go to **Step 6**.

Collect fault information.

Step 6 Log in to FusionInsight Manager, and choose **O&M** > **Log** > **Download**.

Step 7 Select **Spark2x** in the required cluster from the **Service**.

Step 8 Click  in the upper right corner. In the displayed dialog box, set **Start Date** and **End Date** to 10 minutes before and after the alarm generation time respectively and click **OK**. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected logs.

----End

Alarm Clearing

This alarm will be automatically cleared after the fault is rectified.

Related Information

None

10.13.219 ALM-43019 Heap Memory Usage of the IndexServer2x Process Exceeds the Threshold

Description

The system checks the IndexServer2x process status every 30 seconds. The alarm is generated when the heap memory usage of a IndexServer2x process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Severity	Auto Clear
43019	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the available IndexServer2x process heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The heap memory of the IndexServer2x process is overused or the heap memory is inappropriately allocated.

Procedure

Check the heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. In the displayed alarm list, choose the alarm for which the ID is **43019**, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click the IndexServer2x that reported the alarm to go to the **Dashboard** page. Click the drop-down list in the upper right corner of the chart area, and choose **Customize** > **Memory** > **IndexServer2x Memory Usage Statistics** > **OK**. Check whether the heap memory used by the IndexServer2x process reaches the maximum heap memory threshold (95% by default).

- If the threshold is reached, go to **Step 3**.
- If the threshold is not reached, go to **Step 7**.

Step 3 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click the IndexServer2x that reported the alarm to go to the **Dashboard** page. Click the drop-down list in the upper right corner of the chart area, and choose **Customize** > **Memory** > **Statistics for the heap memory of the IndexServer2x Process** > **OK**. Based on the alarm generation time, check the values of the used heap memory of the IndexServer2x process in the corresponding period and obtain the maximum value.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configuration** > **IndexServer2x**> **Tuning**. The default value of the **SPARK_DRIVER_MEMORY** parameter is 4 GB. You can change the value based on the ratio of the maximum heap memory used by the IndexServer2x process to the threshold specified by **IndexServer2x Heap Memory Usage Statistics (IndexServer2x)** in the alarm period. If the alarm persists after the parameter value is changed, increase the value by 0.5 times. If the alarm is generated frequently, double the rate.

NOTE

On FusionInsight Manager, you can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **Memory** > **IndexServer2x Heap Memory Usage Statistics (IndexServer2x)** to view the threshold.

Step 5 Restart all IndexServer2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If the alarm is cleared, no further action is required.
- If the alarm is not cleared, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Expand the **Service** drop-down list, and select **Spark2x** for the target cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Reference

None

10.13.220 ALM-43020 Non-Heap Memory Usage of the IndexServer2x Process Exceeds the Threshold

Description

The system checks the IndexServer2x process status every 30 seconds. The alarm is generated when the non-heap memory usage of the IndexServer2x process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Severity	Auto Clear
43020	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the available IndexServer2x process non-heap memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The non-heap memory of the IndexServer2x process is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. In the displayed alarm list, choose the alarm for which the ID is **43020**, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click the IndexServer2x that reported the alarm to go to the **Dashboard** page. Click the drop-down list in the upper right corner of the chart area, and choose **Customize** > **Memory > IndexServer2x Memory Usage Statistics** > **OK**. Check whether the non-heap memory used by the IndexServer2x process reaches the maximum non-heap memory threshold (95% by default).

- If the threshold is reached, go to **Step 3**.
- If the threshold is not reached, go to **Step 7**.

Step 3 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click the IndexServer2x that reported the alarm to go to the **Dashboard** page. Click the drop-down list in the upper right corner of the chart area, and choose **Customize** > **Memory** > **Statistics for the non-heap memory of the IndexServer2x Process** > **OK**. Based on the alarm generation time, check the values of the used non-heap memory of the IndexServer2x process in the corresponding period and obtain the maximum value.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configurations** > **IndexServer2x**> **Tuning**. You can change the value of **XX:MaxMetaspaceSize** in the **spark.driver.extraJavaOptions** parameter based on the ratio of the maximum non-heap memory used by the IndexServer2x process to the threshold specified by **IndexServer2x Non-Heap Memory Usage Statistics (IndexServer2x)** in the alarm period.

NOTE

On FusionInsight Manager, you can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **Memory** > **IndexServer2x Non-Heap Memory Usage Statistics (IndexServer2x)** to view the threshold.

Step 5 Restart all IndexServer2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If the alarm is cleared, no further action is required.
- If the alarm is not cleared, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Expand the **Service** drop-down list, and select **Spark2x** for the target cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Reference

None

10.13.221 ALM-43021 Direct Memory Usage of the IndexServer2x Process Exceeds the Threshold

Description

The system checks the IndexServer2x process status every 30 seconds. The alarm is generated when the direct heap memory usage of a IndexServer2x process exceeds the threshold (95% of the maximum memory).

Attribute

Alarm ID	Severity	Auto Clear
43021	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the available IndexServer2x process direct memory is insufficient, a memory overflow occurs and the service breaks down.

Possible Causes

The direct heap memory of the IndexServer2x process is overused or the direct heap memory is inappropriately allocated.

Procedure

Check direct heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. In the displayed alarm list, choose the alarm for which the ID is **43021**, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click the IndexServer2x that reported the alarm to go to the **Dashboard** page. Click the drop-down list in the upper right corner of the chart area, and choose **Customize** > **Memory** > **IndexServer2x Memory Usage Statistics** > **OK**. Check whether the direct memory used by the IndexServer2x process reaches the maximum direct memory threshold.

- If the threshold is reached, go to **Step 3**.
- If the threshold is not reached, go to **Step 7**.

Step 3 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance**. Click the IndexServer2x that reported the alarm to go to the **Dashboard** page. Click the drop-down list in the upper right corner of the chart area, and choose **Customize** > **Memory** > **Direct Memory of IndexServer2x** > **OK**. Based on the alarm generation time, check the values of the used direct memory of the IndexServer2x process in the corresponding period and obtain the maximum value.

Step 4 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configurations** > **IndexServer2x**> **Tuning**. You can change the value of **XX:MaxDirectMemorySize** (the default value is 512 MB) in the **spark.driver.extraJavaOptions** parameter based on the ratio of the maximum direct memory used by the IndexServer2x process to the threshold specified by **IndexServer2x Direct Memory Usage Statistics (IndexServer2x)** in the alarm period. If the alarm persists after the parameter value is changed, increase the value by 0.5 times. If the alarm is generated frequently, double the rate.

NOTE

On FusionInsight Manager, you can choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **Memory** > **IndexServer2x Direct Memory Usage Statistics (IndexServer2x)** to view the threshold.

Step 5 Restart all IndexServer2x instances.

Step 6 After 10 minutes, check whether the alarm is cleared.

- If the alarm is cleared, no further action is required.

- If the alarm is not cleared, go to **Step 7**.

Collect fault information.

Step 7 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 8 Expand the **Service** drop-down list, and select **Spark2x** for the target cluster.

Step 9 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 10 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Reference

None

10.13.222 ALM-43022 IndexServer2x Process GC Time Exceeds the Threshold

Description

The system checks the GC time of the IndexServer2x process every 60 seconds. This alarm is generated when the detected GC time exceeds the threshold (12 seconds) for three consecutive times. To change the threshold, choose **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **GC Time** > **Total GC time in milliseconds (IndexServer2x)**. This alarm is cleared when the IndexServer2x GC time is shorter than or equal to the threshold.

Attribute

Alarm ID	Severity	Auto Clear
43022	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Parameter	Description
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the GC time exceeds the threshold, IndexServer2x may run in low performance or even unavailable.

Possible Causes

The heap memory of the IndexServer2x process is overused or the heap memory is inappropriately allocated. As a result, GC occurs frequently.

Procedure

Check the GC time.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. In the displayed alarm list, choose the alarm with ID **43022**, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance** and click the IndexServer2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the Chart area and choose **Customize** > **GC Time** > **Garbage Collection (GC) Time of IndexServer2x** from the drop-down list box in the upper right corner and click **OK** to check whether the GC time is longer than the threshold (default value: 12 seconds).

- If the threshold is reached, go to **Step 3**.
- If the threshold is not reached, go to **Step 6**.

Step 3 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configurations** > **IndexServer2x** > **Default**. The default value of the **SPARK_DRIVER_MEMORY** is 4 GB. You can change the value according to the following rules: Increase the value of the **SPARK_DRIVER_MEMORY** parameter 1.5 times to its default value. If this alarm is still generated occasionally after the adjustment, increase the value by 0.5 times. Double the value if the alarm is reported frequently.

Step 4 Restart all IndexServer2x instances.

Step 5 After 10 minutes, check whether the alarm is cleared.

- If the alarm is cleared, no further action is required.
- If the alarm is not cleared, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Spark2x** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Reference

None

10.13.223 ALM-43023 IndexServer2x Process Full GC Number Exceeds the Threshold

Description

The system checks the Full GC number of the IndexServer2x process every 60 seconds. This alarm is generated when the detected Full GC number exceeds the threshold (12) for three consecutive times. You can change the threshold by choosing **O&M** > **Alarm** > **Thresholds** > *Name of the desired cluster* > **Spark2x** > **GC Number** > **Full GC Number of IndexServer2x**. This alarm is cleared when the Full GC number of the IndexServer2x process is less than or equal to the threshold. This alarm is cleared when the Full GC number of the IndexServer2x process is less than or equal to the threshold.

Attribute

Alarm ID	Severity	Auto Clear
43023	Major	Yes

Parameters

Parameter	Description
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.

Parameter	Description
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the GC number exceeds the threshold, IndexServer2x maybe run in low performance or even unavailable.

Possible Causes

The heap memory of the IndexServer2x process is overused or the heap memory is inappropriately allocated. As a result, Full GC occurs frequently.

Procedure

Check the number of Full GCs.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm > Alarms**. In the displayed alarm list, choose the alarm with the ID **43023**, and check the **RoleName** in **Location** and confirm the IP address of **HostName**.

Step 2 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Instance** and click the IndexServer2x for which the alarm is generated to go to the **Dashboard** page. Click the drop-down menu in the chart area and choose **Customize** > **GC Number > Full GC Number of IndexServer2x** from the drop-down list box in the upper right corner and click **OK** to check whether the GC number is larger than the threshold (default value: 12).

- If the threshold is reached, go to **Step 3**.
- If the threshold is not reached, go to **Step 6**.

Step 3 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations** > **All Configurations** > **IndexServer2x** > **Tuning**. The default value of the **SPARK_DRIVER_MEMORY** is 4 GB. You can change the value according to the following rules: If this alarm is generated occasionally, increase the value by 0.5 times. Double the value if the alarm is reported frequently. In the case of large service volume and high service concurrency, you are advised to add instances.

Step 4 Restart all IndexServer2x instances.

Step 5 After 10 minutes, check whether the alarm is cleared.

- If the alarm is cleared, no further action is required.
- If the alarm is not cleared, go to **Step 6**.

Collect fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Spark2x** for the target cluster.

Step 8 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 9 Contact the O&M personnel and send the collected fault logs.

----End

Alarm Clearing

After the fault is rectified, the system automatically clears this alarm.

Reference

None

10.13.224 ALM-44004 Presto Coordinator Resource Group Queuing Tasks Exceed the Threshold

Description

This alarm is generated when the system detects that the number of queuing tasks in a resource group exceeds the threshold. The system queries the number of queuing tasks in a resource group through the JMX interface. You can choose **Components** > **Presto** > **Service Configuration** (switch **Basic** to **All**) > **Presto** > **resource-groups** to configure a resource group. You can choose **Components** > **Presto** > **Service Configuration** (switch **Basic** to **All**) > **Coordinator** > **Customize** > **resourceGroupAlarm** to configure the threshold of each resource group.

Attribute

Alarm ID	Alarm Severity	Auto Clear
44004	Major	Yes

Parameter

Parameter	Description
ServiceName	Service for which the alarm is generated.
RoleName	Role for which the alarm is generated.
HostName	Host for which the alarm is generated.

Impact on the System

If the number of queuing tasks in a resource group exceeds the threshold, a large number of tasks may be in the queuing state. The Presto task time exceeds the expected value. When the number of queuing tasks in a resource group exceeds the maximum number (**maxQueued**) of queuing tasks in the resource group, new tasks cannot be executed.

Possible Causes

The resource group configuration is improper or too many tasks in the resource group are submitted.

Procedure

Step 1 Choose **Components** > **Presto** > **Service Configuration** (switch **Basic** to **All**) > **Presto** > **resource-groups** to adjust the resource group configuration.

Step 2 You can choose **Components** > **Presto** > **Service Configuration** (switch **Basic** to **All**) > **Coordinator** > **Customize** > **resourceGroupAlarm** to modify the threshold of each resource group.

Step 3 Collect fault information.

1. Log in to the cluster node based on the host name in the fault information and query the number of queuing tasks based on **Resource Group** in the additional information on the Presto client.
2. Log in to the cluster node based on the host name in the fault information, view the **/var/log/Bigdata/nodeagent/monitorlog/monitor.log** file, and search for resource group information to view the monitoring collection information of the resource group.
3. Contact the O&M personnel and send the collected logs.

----End

Reference

None

10.13.225 ALM-44005 Presto Coordinator Process GC Time Exceeds the Threshold

Description

The system collects GC time of the Presto Coordinator process every 30 seconds. This alarm is generated when the GC time exceeds the threshold (exceeds 5 seconds for three consecutive times). You can change the threshold by choosing **System** > **Configure Alarm Threshold** > **Service** > **Presto** > **Coordinator** > **Presto Process Garbage Collection Time** > **Garbage Collection Time of the Coordinator Process** on MRS Manager. This alarm is cleared when the Coordinator process GC time is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
44005	Major	Yes

Parameter

Parameter	Description
ServiceName	Service for which the alarm is generated.
RoleName	Role for which the alarm is generated.
HostName	Host for which the alarm is generated.

Impact on the System

If the GC time of the Coordinator process is too long, the Coordinator process running performance will be affected and the Coordinator process will even be unavailable.

Possible Causes

The heap memory of the Coordinator process is overused or inappropriately allocated, causing frequent occurrence of the GC process.

Procedure

Step 1 Check the GC time.

1. Go to the cluster details page and choose **Alarms**.

   NOTE

   For MRS 1.8.10 or earlier, log in to MRS Manager and choose **Alarms**.

2. Select the alarm whose **Alarm ID** is **44005** and then check the role name in **Location** and confirm the IP adress of the instance.
3. Choose **Components** > **Presto** > **Instances** > **Coordinator** (business IP address of the instance for which the alarm is generated) > **Customize** > **Presto Garbage Collection Time**. Click **OK** to view the GC time.
4. Check whether the GC time of the Coordinator process is longer than 5 seconds.
   - If yes, go to **Step 1.5**.
   - If no, go to **Step 2**.
5. Choose **Components** > **Presto** > **Service Configuration**, and switch **Basic** to **All**. Choose **Presto** > **Coordinator**. Increase the value of **-Xmx** (maximum heap memory) in the **JAVA_OPTS** parameter based on the site requirements.

6. Check whether the alarm is cleared.
   - If yes, no further action is required.
   - If no, go to **Step 2**.

Step 2 Collect fault information.

1. On MRS Manager, choose **System** > **Export Log**.
2. Contact the O&M personnel and send the collected logs.

----End

Reference

None

10.13.226 ALM-44006 Presto Worker Process GC Time Exceeds the Threshold

Description

The system collects GC time of the Presto Worker process every 30 seconds. This alarm is generated when the GC time exceeds the threshold (exceeds 5 seconds for three consecutive times). You can change the threshold by choosing **System** > **Configure Alarm Threshold** > **Service** > **Presto** > **Worker** > **Presto Garbage Collection Time** > **Garbage Collection Time of the Worker Process** on MRS Manager. This alarm is cleared when the Worker process GC time is shorter than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
44006	Major	Yes

Parameter

Parameter	Description
ServiceName	Service for which the alarm is generated.
RoleName	Role for which the alarm is generated.
HostName	Host for which the alarm is generated.

Impact on the System

If the GC time of the Worker process is too long, the Worker process running performance will be affected and the Worker process will even be unavailable.

Possible Causes

The heap memory of the Worker process is overused or inappropriately allocated, causing frequent occurrence of the GC process.

Procedure

Step 1 Check the GC time.

1. Go to the cluster details page and choose **Alarms**.

   NOTE

   For MRS 1.8.10 or earlier, log in to MRS Manager and choose **Alarms**.

2. Select the alarm whose **Alarm ID** is **44006**. Then check the role name in **Location** and confirm the IP adress of the instance.
3. Choose **Components** > **Presto** > **Instances** > **Worker** (business IP address of the instance for which the alarm is generated) > **Customize** > **Presto Garbage Collection Time**. Click **OK** to view the GC time.
4. Check whether the GC time of the Worker process is longer than 5 seconds.
   - If yes, go to **Step 1.5**.
   - If no, go to **Step 2**.
5. Choose **Components** > **Presto** > **Service Configuration**, and switch **Basic** to **All**, and choose **Presto** > **Worker** Increase the value of **-Xmx** (maximum heap memory) in the **JAVA_OPTS** parameter based on the site requirements.
6. Check whether the alarm is cleared.
   - If yes, no further action is required.
   - If no, go to **Step 2**.

Step 2 Collect fault information.

1. On MRS Manager, choose **System** > **Export Log**.
2. Contact the O&M personnel and send the collected logs.

----End

Reference

None

10.13.227 ALM-45175 Average Time for Calling OBS Metadata APIs Is Greater than the Threshold

Description

The system checks whether the average duration for calling OBS metadata APIs is greater than the threshold every 30 seconds. This alarm is generated when the number of consecutive times that the average time exceeds the specified threshold is greater than the number of smoothing times.

This alarm is automatically cleared when the average duration for calling the OBS metadata APIs is lower than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45175	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the average time for calling the OBS metadata APIs exceeds the threshold, the upper-layer big data computing services may be affected. To be more specific, the execution time of some computing tasks will exceed the threshold.

Possible Causes

Frame freezing occurs on the OBS server, or the network between the OBS client and the OBS server is unstable.

Procedure

Check the heap memory usage.

Step 1 On the **FusionInsight Manager** homepage, choose **O&M** > **Alarm** > **Alarms** > **Average Time for Calling the OBS Metadata API Exceeds the Threshold**, view the role name in **Location**, and check the instance IP address.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **meta** > **Instance** > **meta** (IP address of the instance for which the alarm is generated). Click the drop-down list in the upper right corner of the chart area and choose **Customize**. In the dialog box that is displayed, select **Average time of OBS interface calls** from **OBS Meta data Operations**, and click **OK**. Check whether the average time of OBS metadata API calls exceeds the threshold.

- If yes, go to **Step 3**.

- If no, go to **Step 5**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **O&M** > **Alarm** > **Thresholds** > **meta** > **Average Time for Calling the OBS Metadata API**. Increase the threshold or smoothing times as required.

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 In the **Services** area, select **NodeAgent**, **NodeMetricAgent**, **OmmServer**, and **OmmAgent** under OMS.

Step 7 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.228 ALM-45176 Success Rate of Calling OBS Metadata APIs Is Lower than the Threshold

Description

The system checks whether the success rate of calling OBS metadata APIs is lower than the threshold every 30 seconds. This alarm is generated when the success rate is lower than the threshold.

This alarm is automatically cleared when the success rate of calling APIs for writing OBS data is greater than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45176	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the success rate of calling the OBS metadata APIs is less than the threshold, the upper-layer big data computing services may be affected. To be more specific, some computing tasks may fail to be executed.

Possible Causes

An execution exception or severe timeout occurs on the OBS server.

Procedure

Check the heap memory usage.

Step 1 On the **FusionInsight Manager** homepage, choose **O&M** > **Alarm** > **Alarms** > **Success Rate for Calling the OBS Metadata API Is Lower Than the Threshold**, view the role name in **Location**, and check the instance IP address.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **meta** > **Instance** > **meta** (IP address of the instance for which the alarm is generated). Click the drop-down list in the upper right corner of the chart area and choose **Customize**. In the dialog box that is displayed, select **Success percent of OBS interface calls** from **OBS Meta data Operations**, and click **OK**. Check whether the average time of OBS metadata API calls exceeds the threshold.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **O&M** > **Alarm** > **Thresholds** > **meta** > **Success Rate for Calling the OBS Metadata API**. Increase the threshold or smoothing times as required.

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.

- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 In the **Services** area, select **NodeAgent**, **NodeMetricAgent**, **OmmServer**, and **OmmAgent** under OMS.

Step 7 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.229 ALM-45177 Success Rate of Calling OBS Data Read APIs Is Lower than the Threshold

Description

The system checks whether the success rate of calling APIs for reading OBS data is lower than the threshold every 30 seconds. This alarm is generated when the success rate is lower than the threshold.

This alarm is automatically cleared when the success rate of calling APIs for reading OBS data is greater than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45177	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.

Name	Meaning
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the success rate of calling the OBS APIs for reading data is less than the threshold, the upper-layer big data computing services may be affected. To be more specific, some computing tasks may fail to be executed.

Possible Causes

An execution exception or severe timeout occurs on the OBS server.

Procedure

Check the heap memory usage.

Step 1 On the **FusionInsight Manager** homepage, choose **O&M** > **Alarm** > **Alarms** > **Success Rate for Calling the OBS Data Read API Is Lower Than the Threshold**, view the role name in **Location**, and check the instance IP address.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **meta** > **Instance** > **meta** (IP address of the instance for which the alarm is generated). Click the drop-down list in the upper right corner of the chart area and choose **Customize**. In the dialog box that is displayed, select **Success percent of OBS data read operation interface calls** from **OBS data read operation**, and click **OK**. Check whether the average time of OBS metadata API calls exceeds the threshold.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **O&M** > **Alarm** > **Thresholds** > **meta** > **Success Rate for Calling the OBS Data Read API**. Increase the threshold or smoothing times as required.

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 In the **Services** area, select **NodeAgent**, **NodeMetricAgent**, **OmmServer**, and **OmmAgent** under OMS.

Step 7 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.230 ALM-45178 Success Rate of Calling OBS Data Write APIs Is Lower Than the Threshold

Description

The system checks whether the success rate of calling APIs for writing OBS data is lower than the threshold every 30 seconds. This alarm is generated when the success rate is lower than the threshold.

This alarm is automatically cleared when the success rate of calling APIs for writing OBS data is greater than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45178	Minor	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

If the success rate of calling the OBS APIs for writing data is lower than the threshold, the upper-layer big data computing services may be affected. To be more specific, some computing tasks may fail to be executed.

Possible Causes

An execution exception or severe timeout occurs on the OBS server.

Procedure

Check the heap memory usage.

Step 1 On the **FusionInsight Manager** homepage, choose **O&M** > **Alarm** > **Alarms** > **Success Rate for Calling the OBS Data Write API Is Lower Than the Threshold**, view the role name in **Location**, and check the instance IP address.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **meta** > **Instance** > **meta** (IP address of the instance for which the alarm is generated). Click the drop-down list in the upper right corner of the chart area and choose **Customize**. In the dialog box that is displayed, select **Success percent of OBS data write operation interface calls** from **OBS data write operation**, and click **OK**. Check whether the average time of OBS metadata API calls exceeds the threshold.

- If yes, go to **Step 3**.
- If no, go to **Step 5**.

Step 3 Choose **Cluster** > *Name of the desired cluster* > **O&M** > **Alarm** > **Thresholds** > **meta** > **Success Rate for Calling the OBS Data Write API**. Increase the threshold or smoothing times as required.

Step 4 Check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 In the **Services** area, select **NodeAgent**, **NodeMetricAgent**, **OmmServer**, and **OmmAgent** under OMS.

Step 7 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 30 minutes ahead of and after the alarm generation time respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.231 ALM-45275 Ranger Service Unavailable

Description

The alarm module checks the Ranger service status every 180 seconds. This alarm is generated if the Ranger service is abnormal.

This alarm is cleared after the Ranger service recovers.

Attributes

Alarm ID	Alarm Severity	Automatically Cleared
45275	Critical	Yes

Parameters

Name	Meaning
Source	Cluster for which the alarm is generated.
ServiceName	Service for which the alarm is generated.
RoleName	Role for which the alarm is generated.
HostName	Host for which the alarm is generated.

Impact on the System

When the Ranger service is unavailable, Ranger cannot work properly and the native Ranger UI cannot be accessed.

Possible Causes

- The DBService service on which Ranger depends is abnormal.
- The RangerAdmin role instance is abnormal.

Procedure

Check the DBService process status.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. On the displayed page, check whether the ALM-27001 DBService Service Unavailable alarm is reported.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Rectify the DBService service fault by following the handling procedure of ALM-27001 DBService Service Unavailable. After the DBService alarm is cleared, check whether Ranger Service Unavailable alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Check all RangerAdmin instances.

Step 3 Log in to the node where the RangerAdmin instance is located as user **omm** and run the **ps -ef|grep "proc_rangeradmin"** command to check whether the RangerAdmin process exists on the current node.

- If yes, go to **Step 5**.
- If no, restart the faulty RangerAdmin instance or Ranger service and go to **Step 4**.

Step 4 In the alarm list, check whether the alarm "Ranger Service Unavailable" is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 6 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 7 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.232 ALM-45276 Abnormal RangerAdmin status

Description

The alarm module checks the RangerAdmin status every 60 seconds. This alarm is generated when the RangerAdmin status is abnormal.

This alarm is cleared after the RangerAdmin status recovers.

Attributes

Alarm ID	Alarm Severity	Automatically Cleared
45276	Major	Yes

Parameters

Name	Meaning
Source	Cluster for which the alarm is generated.
ServiceName	Service for which the alarm is generated.
RoleName	Role for which the alarm is generated.
HostName	Host for which the alarm is generated.

Impact on the System

If the status of a single RangerAdmin is abnormal, the access to the Ranger native UI is not affected. When the status of two RangerAdmins is abnormal, the native UI of Ranger cannot be accessed, and policies cannot be created, modified, or deleted.

Possible Causes

The RangerAdmin port is not started.

Procedure

Check the port process.

Step 1 In the alarm list on FusionInsight Manager, locate the row that contains the alarm, and click ⌄ to query the host name.

Step 2 Log in to the node where the abnormal RangerAdmin instance resides as user **omm**. Run the **ps -ef|grep "proc_rangeradmin" | grep -v grep | awk -F ' ' '{print $2}'** command to obtain ***pid*** of the RangerAdmin process, and run the **netstat -anp|grep *pid* | grep LISTEN** command to check whether the RangerAdmin process listens to port 21401 for a cluster in security mode and port 21400 for a cluster in normal mode.

- If yes, go to **Step 4**.
- If no, restart the faulty RangerAdmin instance or Ranger service and go to **Step 3**.

Step 3 In the alarm list, check whether the **Abnormal RangerAdmin status** alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 4**.

Collect the fault information.

Step 4 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 5 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 6 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 7 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.233 ALM-45277 RangerAdmin Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of the RangerAdmin service every 60 seconds. This alarm is generated when the system detects that the heap memory usage of the RangerAdmin instance exceeds the threshold (95% of the maximum memory) for 10 consecutive times. This alarm is cleared when the heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45277	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Heap memory overflow may cause service breakdown.

Possible Causes

The heap memory usage of the RangerAdmin instance is high or the heap memory is improperly allocated.

Procedure

Check the heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45277 RangerAdmin Heap Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **RangerAdmin Heap Memory Usage**. Click **OK**.

Step 3 Check whether the heap memory used by RangerAdmin reaches the threshold (95% of the maximum heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **RangerAdmin** > **Instance Configuration**. Click **All Configurations**, and choose **RangerAdmin** > **System**. Increase the value of **-Xmx** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for RangerAdmin cannot meet the heap memory required by the RangerAdmin process. You are advised to check the heap memory usage of RangerAdmin and change the value of **-Xmx** in **GC_OPTS** to the twice of the heap memory used by RangerAdmin. The value can be changed based on the actual service scenario. For details, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.234 ALM-45278 RangerAdmin Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the RangerAdmin service every 60 seconds. This alarm is generated when the direct memory usage of the RangerAdmin instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the direct memory usage of RangerAdmin is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45278	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Direct memory overflow may cause service breakdown.

Possible Causes

The direct memory of the RangerAdmin instance is overused or the direct memory is inappropriately allocated. As a result, the memory usage exceeds the threshold.

Procedure

Check the direct memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45278 RangerAdmin Direct Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **RangerAdmin Direct Memory Usage**. Click **OK**.

Step 3 Check whether the direct memory used by RangerAdmin reaches the threshold (80% of the maximum direct memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **RangerAdmin** > **Instance Configuration**. Click **All Configurations**, and choose **RangerAdmin** > **System**. Increase the value of **-XX:MaxDirectMemorySize** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the direct memory configured for RangerAdmin cannot meet the direct memory required by the RangerAdmin process. You are advised to check the direct memory usage of RangerAdmin and change the value of **-XX:MaxDirectMemorySize** in **GC_OPTS** to the twice of the direct memory used by RangerAdmin. You can change the value based on the actual service scenario. For details, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.235 ALM-45279 RangerAdmin Non Heap Memory Usage Exceeds the Threshold

Description

The system checks the non-heap memory usage of the RangerAdmin service every 60 seconds. This alarm is generated when the non-heap memory usage of the RangerAdmin instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the non-heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45279	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Non-heap memory overflow may cause service breakdown.

Possible Causes

The non-heap memory usage of the RangerAdmin instance is high or the non-heap memory is improperly allocated.

Procedure

Check non-heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45279 RangerAdmin Non Heap Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **RangerAdmin Non Heap Memory Usage**. Click **OK**.

Step 3 Check whether the non-heap memory used by RangerAdmin reaches the threshold (80% of the maximum non-heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **RangerAdmin** > **Instance Configuration**. Click **All Configurations**, and choose **RangerAdmin** > **System**. Set **-XX: MaxPermSize** in the **GC_OPTS** parameter to a larger value based on site requirements and save the configuration.

NOTE

If this alarm is generated, the non-heap memory size configured for the RangerAdmin instance cannot meet the non-heap memory required by the RangerAdmin process. You are advised to change the value of **-XX:MaxPermSize** in **GC_OPTS** to the twice of the current non-heap memory usage or change the value based on the site requirements.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.236 ALM-45280 RangerAdmin GC Duration Exceeds the Threshold

Description

The system checks the GC duration of the RangerAdmin process every 60 seconds. This alarm is generated when the GC duration of the RangerAdmin process exceeds the threshold (12 seconds by default) for five consecutive times. This alarm is cleared when the GC duration is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45280	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

The RangerAdmin responds slowly.

Possible Causes

The heap memory of the RangerAdmin instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45280 RangerAdmin GC Duration Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated and click the drop-down list in the upper right corner of the chart area. Choose **Customize** > **GC** > **RangerAdmin GC Duration**. Click **OK**.

Step 3 Check whether the GC duration of the RangerAdmin process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.

- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **RangerAdmin** > **Instance Configuration**. Click **All Configurations**, and choose **RangerAdmin** > **System**. Increase the value of **-Xmx** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for RangerAdmin cannot meet the heap memory required by the RangerAdmin process. You are advised to check the heap memory usage of RangerAdmin and change the value of **-Xmx** in **GC_OPTS** to the twice of the heap memory used by RangerAdmin. The value can be changed based on the actual service scenario. For details, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.237 ALM-45281 UserSync Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of the UserSync service every 60 seconds. This alarm is generated when the system detects that the heap memory usage of the UserSync instance exceeds the threshold (95% of the maximum memory) for 10 consecutive times. This alarm is cleared when the heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45281	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Heap memory overflow may cause service breakdown.

Possible Causes

The heap memory usage of the UserSync instance is high or the heap memory is improperly allocated.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45281 UserSync Heap Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **UserSync Heap Memory Usage**. Click **OK**.

Step 3 Check whether the heap memory used by UserSync reaches the threshold (95% of the maximum heap memory by default).

- If yes, go to **Step 4**.

- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **UserSync** > **Instance Configuration**. Click **All Configurations**, and choose **UserSync** > **System**. Increase the value of **-Xmx** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for UserSync cannot meet the heap memory required by the UserSync process. You are advised to change the **-Xmx** value of **GC_OPTS** to twice that of the heap memory used by UserSync. You can change the value based on the actual service scenario. For details about how to check the UserSync heap memory usage, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.238 ALM-45282 UserSync Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the UserSync service every 60 seconds. This alarm is generated when the direct memory usage of the UserSync instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the UserSync direct memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45282	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Direct memory overflow may cause service breakdown.

Possible Causes

The direct memory of the UserSync instance is overused or the direct memory is inappropriately allocated. As a result, the memory usage exceeds the threshold.

Procedure

Check the direct memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45282 UserSync Direct Memory Usage Exceeds the Threshold**. Check the location information of the alarm. Check the name of the instance host for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **UserSync Direct Memory Usage**. Click **OK**.

Step 3 Check whether the direct memory used by the UserSync reaches the threshold (80% of the maximum direct memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **UserSync** > **Instance Configuration**. Click **All Configurations**, and choose **UserSync** > **System**. Increase the value of **-XX:MaxDirectMemorySize** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the direct memory configured for UserSync cannot meet the direct memory required by the UserSync process. You are advised to check the direct memory usage of UserSync and change the value of **-XX:MaxDirectMemorySize** in **GC_OPTS** to the twice of the direct memory used by UserSync. You can change the value based on the actual service scenario. For details, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.239 ALM-45283 UserSync Non Heap Memory Usage Exceeds the Threshold

Description

The system checks the non-heap memory usage of the UserSync service every 60 seconds. This alarm is generated when the non-heap memory usage of the UserSync instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the non-heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45283	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Non-heap memory overflow may cause service breakdown.

Possible Causes

The non-heap memory of the UserSync process is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45283 UserSync Non Heap Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **UserSync Non Heap Memory Usage**. Click **OK**.

Step 3 Check whether the non-heap memory used by UserSync reaches the threshold (80% of the maximum non-heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **UserSync** > **Instance Configuration**. Click **All Configurations**, and choose **UserSync** > **System**. Set **-XX: MaxPermSize** in the **GC_OPTS** parameter to a larger value based on site requirements and click **Save** to save the configuration.

NOTE

If this alarm is generated, the non-heap memory size configured for the UserSync instance cannot meet the non-heap memory required by the UserSync process. You are advised to change the **-XX:MaxPermSize** value of **GC_OPTS** to twice that of the current non-heap memory size or change the value based on the site requirements.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.240 ALM-45284 UserSync Garbage Collection (GC) Time Exceeds the Threshold

Description

The system checks the GC duration of the UserSync process every 60 seconds. This alarm is generated when the GC duration of the UserSync process exceeds the threshold (12 seconds by default) for five consecutive times. This alarm is cleared when the GC duration is less than the threshold.

Attributes

Alarm ID	Alarm Severity	Automatically Cleared
45284	Major	Yes

Parameters

Name	Meaning
Source	Cluster for which the alarm is generated.
ServiceName	Service for which the alarm is generated.
RoleName	Role for which the alarm is generated.
HostName	Host for which the alarm is generated.
Trigger Condition	Threshold for triggering the alarm.

Impact on the System

UserSync responds slowly.

Possible Causes

The heap memory of the UserSync instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC time.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45284 UserSync Garbage Collection (GC) Time Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated and click the drop-down list in the upper right corner of the chart area. Choose **Customize** > **GC** > **UserSync GC Duration**. Click **OK**.

Step 3 Check whether the GC duration of the UserSync process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **UserSync** > **Instance Configuration**. Click **All Configurations**, and choose

UserSync > **System**. Increase the value of **-Xmx** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for UserSync cannot meet the heap memory required by the UserSync process. You are advised to change the **-Xmx** value of **GC_OPTS** to the twice that of the heap memory used by UserSync. You can change the value based on the actual service scenario. For details about how to check the UserSync heap memory usage, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M** > **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

After the fault that triggers the alarm is rectified, the alarm is automatically cleared.

Related Information

None

10.13.241 ALM-45285 TagSync Heap Memory Usage Exceeds the Threshold

Description

The system checks the heap memory usage of the TagSync service every 60 seconds. This alarm is generated when the heap memory usage of the TagSync instance exceeds the threshold (95% of the maximum memory) for 10 consecutive times. This alarm is cleared when the heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45285	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Heap memory overflow may cause service breakdown.

Possible Causes

The heap memory usage of the TagSync instance is high or the heap memory is improperly allocated.

Procedure

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45285 TagSync Heap Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **TagSync Heap Memory Usage**. Click **OK**.

Step 3 Check whether the heap memory used by TagSync reaches the threshold (95% of the maximum heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **TagSync** > **Instance Configuration**. Click **All Configurations** and choose **TagSync** > **System**. Increase the value of **-Xmx** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for TagSync cannot meet the heap memory required by the TagSync process. You are advised to change the **-Xmx** value of **GC_OPTS** to twice that of the heap memory used by TagSync. You can change the value based on the actual service scenario. For details about how to check the TagSync heap memory usage, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.242 ALM-45286 TagSync Direct Memory Usage Exceeds the Threshold

Description

The system checks the direct memory usage of the TagSync service every 60 seconds. This alarm is generated when the direct memory usage of the TagSync instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the TagSync direct memory usage is less than or equal to the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45286	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Direct memory overflow may cause service breakdown.

Possible Causes

The direct memory of the TagSync instance is overused or the direct memory is inappropriately allocated. As a result, the memory usage exceeds the threshold.

Procedure

Check the direct memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45286 TagSync Direct Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **TagSync Direct Memory Usage**. Click **OK**.

Step 3 Check whether the direct memory used by the TagSync reaches the threshold (80% of the maximum direct memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **TagSync** > **Instance Configuration**. Click **All Configurations** and choose **TagSync** > **System**. Increase the value of **-XX:MaxDirectMemorySize** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the direct memory configured for TagSync cannot meet the direct memory required by the TagSync process. You are advised to check the direct memory usage of TagSync and change the value of **-XX:MaxDirectMemorySize** in **GC_OPTS** to the twice of the direct memory used by TagSync. You can change the value based on the actual service scenario. For details, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.243 ALM-45287 TagSync Non Heap Memory Usage Exceeds the Threshold

Description

The system checks the non-heap memory usage of the TagSync service every 60 seconds. This alarm is generated when the non-heap memory usage of the TagSync instance exceeds the threshold (80% of the maximum memory) for five consecutive times. This alarm is cleared when the non-heap memory usage is less than the threshold.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45287	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

Non-heap memory overflow may cause service breakdown.

Possible Causes

The non-heap memory of the TagSync process is overused or the non-heap memory is inappropriately allocated.

Procedure

Check non-heap memory usage.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45287 TagSync Non Heap Memory Usage Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated. Click the drop-down list in the upper right corner of the chart area and choose **Customize** > **CPU and Memory** > **TagSync Non Heap Memory Usage**. Click **OK**.

Step 3 Check whether the non-heap memory used by TagSync reaches the threshold (80% of the maximum non-heap memory by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **TagSync** > **Instance Configuration**. Click **All Configurations** and choose **TagSync** > **System**. Set **-XX: MaxPermSize** in the **GC_OPTS** parameter to a larger value based on site requirements and save the configuration.

NOTE

If this alarm is generated, the non-heap memory size configured for the TagSync instance cannot meet the non-heap memory required by the TagSync process. You are advised to change the **-XX:MaxPermSize** value of **GC_OPTS** to twice that of the current non-heap memory size or change the value based on the site requirements.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.244 ALM-45288 TagSync Garbage Collection (GC) Time Exceeds the Threshold

Description

The system checks the GC duration of the TagSync process every 60 seconds. This alarm is generated when the GC duration of the TagSync process exceeds the threshold (12 seconds by default) for five consecutive times. This alarm is cleared when the GC duration is less than the threshold.

Attributes

Alarm ID	Alarm Severity	Auto Clear
45288	Major	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.
Trigger Condition	Specifies the threshold for triggering the alarm.

Impact on the System

TagSync responds slowly.

Possible Causes

The heap memory of the TagSync instance is overused or the heap memory is inappropriately allocated. As a result, GCs occur frequently.

Procedure

Check the GC duration.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms** > **ALM-45288 TagSync Garbage Collection (GC) Time Exceeds the Threshold**. Check the location information of the alarm and view the host name of the instance for which the alarm is generated.

Step 2 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance**. Select the role corresponding to the host name of the instance for which the alarm is generated and click the drop-down list in the upper right corner of the chart area. Choose **Customize** > **GC** > **TagSync GC Duration**. Click **OK**.

Step 3 Check whether the GC duration of the TagSync process collected every minute exceeds the threshold (12 seconds by default).

- If yes, go to **Step 4**.
- If no, go to **Step 6**.

Step 4 On FusionInsight Manager, choose **Cluster** > **Services** > **Ranger** > **Instance** > **TagSync** > **Instance Configuration**. Click **All Configurations** and choose **TagSync** > **System**. Increase the value of **-Xmx** in the **GC_OPTS** parameter based on the site requirements and save the configuration.

NOTE

If this alarm is generated, the heap memory configured for TagSync cannot meet the heap memory required by the TagSync process. You are advised to change the **-Xmx** value of **GC_OPTS** to twice that of the heap memory used by TagSync. You can change the value based on the actual service scenario. For details about how to check the TagSync heap memory usage, see **Step 2**.

Step 5 Restart the affected services or instances and check whether the alarm is cleared.

- If yes, no further action is required.
- If no, go to **Step 6**.

Collect the fault information.

Step 6 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 7 Expand the **Service** drop-down list, and select **Ranger** for the target cluster.

Step 8 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 10 minutes ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 9 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.245 ALM-45425 ClickHouse Service Unavailable

Description

The alarm module checks the ClickHouse instance status every 60 seconds. This alarm is generated when the alarm module detects that all ClickHouse instances are abnormal.

This alarm is cleared when the system detects that any ClickHouse instance is restored and the alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45425	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

The ClickHouse service is abnormal. You cannot use FusionInsight Manager to perform cluster operations on the ClickHouse service. The ClickHouse service function is unavailable.

Possible Causes

The configuration information in the **metrika.xml** file in the component configuration directory of the faulty ClickHouse instance node is inconsistent with that of the corresponding ClickHouse instance in the ZooKeeper.

Procedure

Check whether the configuration in metrika.xml of the ClickHouse instance is correct.

Step 1 Log in to FusionInsight Manager, choose **Cluster** > **Services** > **ClickHouse** > **Instance**, and locate the abnormal ClickHouse instance based on the alarm information.

- If yes, go to **Step 2**.
- If no, go to **Step 9**.

Step 2 Log in to the host where the ClickHouse service is abnormal and ping the IP address of another normal ClickHouse instance node to check whether the network connection is normal.

- If yes, go to **Step 3**.
- If no, contact the network administrator to repair the network.

Step 3 Choose **Cluster** > **Services** > **ClickHouse** > **Instance**, click the abnormal instance name in the **Role** column, click **Configurations**, search for **macros.id** in the search box, and find the value of **macros.id** of the current instance.

Step 4 Log in to the host where the ZooKeeper client is located and log in to the ZooKeeper client.

Switch to the client installation directory.

Example: **cd /opt/client**

Run the following command to configure environment variables:

source bigdata_env

Run the following command to authenticate the user (skip this step in common mode):

kinit *Component service user*

Run the following command to log in to the client tool:

zkCli.sh -server *service IP address of the node where the ZooKeeper role instance locates*:*client port*

Step 5 Run the following command to check whether the ClickHouse cluster topology information can be obtained.

get /clickhouse/config/*value of* ***macros.id*** *in* **Step 3**/**metrika.xml**

- If yes, go to **Step 6**.
- If no, go to **Step 9**.

Step 6 Log in to the host where the ClickHouse instance is abnormal and go to the configuration directory of the ClickHouse instance.

cd ${BIGDATA_HOME}/**FusionInsight_ClickHouse_***Version*/x_x_**ClickHouseServer/etc**

cat metrika.xml

Step 7 Check whether the cluster topology information on ZooKeeper obtained in **Step 5** is the same as that in the **metrika.xml** file in the component configuration directory in **Step 6**.

- If yes, check whether the alarm is cleared. If the alarm persists, go to **Step 9**.
- If no, go to **Step 8**.

Step 8 On FusionInsight Manager, choose **Cluster** > **Services** > **ClickHouse**, click **More**, and select **Synchronize Configuration**. Then, check whether the service status is normal and whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, go to **Step 9**.

Collect the fault information.

Step 9 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 10 Expand the **Service** drop-down list, and select **ClickHouse** for the target cluster.

Step 11 Click in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 12 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.246 ALM-45426 ClickHouse Service Quantity Quota Usage in ZooKeeper Exceeds the Threshold

Description

The alarm module checks the quota usage of the ClickHouse service in the ZooKeeper every 60 seconds. This alarm is generated when the alarm module detects that the usage exceeds the threshold (90%).

This alarm is cleared when the system detects that the usage is lower than the threshold and the alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45426	Major (default)	Yes

Parameters

Name	Meaning
Source	Specifies the cluster or system for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

After the ZooKeeper quantity quota of the ClickHouse service exceeds the threshold, you cannot perform cluster operations on the ClickHouse service on FusionInsight Manager. As a result, the ClickHouse service cannot be used.

Possible Causes

When table data is created, inserted, or deleted, the ClickHouse creates znodes on ZooKeeper nodes. As the service volume increases, the number of znodes may exceed the configured threshold.

Procedure

Check the number of znodes created by ClickHouse on ZooKeeper.

Step 1 Log in to the host where the ZooKeeper client is located and log in to the ZooKeeper client.

Switch to the client installation directory.

Example: **cd /opt/client**

Run the following command to configure environment variables:

source bigdata_env

Run the following command to authenticate the user (skip this step in common mode):

kinit *Component service user*

Run the following command to log in to the client tool:

zkCli.sh -server *service IP address of the node where the ZooKeeper role instance locates*:*client port*

Step 2 Run the following command to check the quota used by ClickHouse on ZooKeeper and check whether the ratio of the **count** value of **Output stat** to the **count** value of **Output quota** in the command output is greater than **0.9**.

listquota /clickhouse

```
absolute path is /zookeeper/quota/clickhouse
Output quota for /clickhouse count=200000,bytes=1000000000
Output stat for /clickhouse count=2667,bytes=60063
```

In the preceding information, the **count** value of **Output stat** is **2667**, and the **count** value of **Output quota** is **200000**.

- If yes, go to **Step 4**.
- If no, check whether the alarm is cleared 5 minutes later. If the alarm persists, go to **Step 5**.

Step 3 On FusionInsight Manager, choose **Cluster** > **Services** > **ClickHouse** > **Configurations** > **All Configurations**, search for the **clickhouse.zookeeper.quota.node.count** parameter, and change the value of this parameter to twice the **count** value of **Output stat** in **Step 2**.

Step 4 Restart the ClickHouse instance for which the alarm is generated, and check whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, perform **Step 4** again, and check whether the alarm is cleared 5 minutes later. If the alarm persists, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 Expand the **Service** drop-down list, and select **ClickHouse** for the target cluster.

Step 7 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.247 ALM-45427 ClickHouse Service Capacity Quota Usage in ZooKeeper Exceeds the Threshold

Description

The alarm module checks the quota usage of the ClickHouse service in the ZooKeeper every 60 seconds. This alarm is generated when the alarm module detects that the usage exceeds the threshold (90%).

This alarm is cleared when the system detects that the usage is lower than the threshold and the alarm is cleared.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45427	Major (default)	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.

Name	Meaning
HostName	Specifies the host for which the alarm is generated.

Impact on the System

After the ZooKeeper quantity quota of the ClickHouse service exceeds the threshold, you cannot perform cluster operations on the ClickHouse service on FusionInsight Manager. As a result, the ClickHouse service cannot be used.

Possible Causes

When table data is created, inserted, or deleted, the ClickHouse creates znodes on ZooKeeper nodes. As the service volume increases, the capacity of znodes may exceed the configured threshold.

Procedure

Check the znode capacity of the ClickHouse in the ZooKeeper.

Step 1 Log in to the host where the ZooKeeper client is located and log in to the ZooKeeper client.

Switch to the client installation directory.

Example: **cd /opt/client**

Run the following command to configure environment variables:

source bigdata_env

Run the following command to authenticate the user (skip this step in common mode):

kinit *Component service user*

Run the following command to log in to the client tool:

zkCli.sh -server *service IP address of the node where the ZooKeeper role instance locates*:*client port*

Step 2 Run the following command to check the quota used by ClickHouse on ZooKeeper and check whether the ratio of the **bytes** value of **Output stat** to the **bytes** value of **Output quota** in the command output is greater than **0.9**.

listquota /clickhouse

```
absolute path is /zookeeper/quota/clickhouse
Output quota for /clickhouse count=200000,bytes=1000000000
Output stat for /clickhouse count=2667,bytes=60063
```

In the preceding information, the **bytes** value of **Output stat** is **60063**, and the **bytes** value of **Output quota** is **1000000000**.

- If yes, go to **Step 4**.
- If no, check whether the alarm is cleared 5 minutes later. If the alarm persists, go to **Step 5**.

Step 3 On FusionInsight Manager, choose **Cluster** > **Services** > **ClickHouse** > **Configurations** > **All Configurations**, search for the **clickhouse.zookeeper.quota.size** parameter, and change the value of this parameter to twice the **bytes** value of **Output stat** in **Step 2**.

Step 4 Restart the ClickHouse instance for which the alarm is generated, and check whether the alarm is cleared 5 minutes later.

- If yes, no further action is required.
- If no, perform **Step 4** again, and check whether the alarm is cleared 5 minutes later. If the alarm persists, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 Expand the **Service** drop-down list, and select **ClickHouse** for the target cluster.

Step 7 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

10.13.248 ALM-45736 Guardian Service Unavailable

Description

The alarm module checks the Guardian service status every 60 seconds. This alarm is generated if Guardian is unavailable.

This alarm is cleared after Guardian recovers.

Attribute

Alarm ID	Alarm Severity	Auto Clear
45275	Critical	Yes

Parameters

Name	Meaning
Source	Specifies the cluster for which the alarm is generated.
ServiceName	Specifies the service for which the alarm is generated.
RoleName	Specifies the role for which the alarm is generated.
HostName	Specifies the host for which the alarm is generated.

Impact on the System

Guardian cannot work properly.

Possible Causes

- The Ranger or HDFS service on which Guardian depends is abnormal.
- The TokenServer role instance is abnormal.

Procedure

Check the Ranger and HDFS service status.

Step 1 On FusionInsight Manager, choose **O&M** > **Alarm** > **Alarms**. On the displayed page, check whether ALM-45275 Ranger Service Unavailable or ALM-14000 HDFS Service Unavailable is reported.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Handle the alarm by referring to section "ALM-45275 Ranger Service Unavailable" or "ALM-14000 HDFS Service Unavailable".

After the alarm is cleared, wait a few minutes and check whether the alarm GuardianService Unavailable is cleared.

- If yes, no further action is required.
- If no, go to **Step 3**.

Check all TokenServer instances.

Step 3 Log in to the node where the TokenServer instance is located as user **omm** and run the **ps -ef|grep "ranger-obs-service"** command to check whether the TokenServer process exists on the node.

- If yes, go to **Step 5**.
- If no, restart the faulty TokenServer instance and go to **Step 4**.

Step 4 In the alarm list, check whether the alarm "Guardian Service Unavailable" is cleared.

- If yes, no further action is required.
- If no, go to **Step 5**.

Collect the fault information.

Step 5 On FusionInsight Manager, choose **O&M**. In the navigation pane on the left, choose **Log** > **Download**.

Step 6 Expand the **Service** drop-down list, and select **Guardian** for the target cluster.

Step 7 Click  in the upper right corner, and set **Start Date** and **End Date** for log collection to 1 hour ahead of and after the alarm generation time, respectively. Then, click **Download**.

Step 8 Contact O&M personnel and provide the collected logs.

----End

Alarm Clearing

This alarm is automatically cleared after the fault is rectified.

Related Information

None

11 MRS Manager Operation Guide (Applicable to 2.x and Earlier Versions)

11.1 Introduction to MRS Manager

Overview

MRS manages and analyzes massive data and helps you rapidly obtain desired data from structured and unstructured data. The structure of open-source components is complex. The installation, configuration, and management processes are time- and labor-consuming. MRS Manager is a unified enterprise-level cluster management platform and provides the following functions:

- Cluster monitoring enables you to quickly view the health status of hosts and services.
- Graphical metric monitoring and customization enable you to quickly obtain key information about the system.
- Service property configurations can meet service performance requirements.
- With cluster, service, and role instance functions, you can start or stop services and clusters in one click.

Introduction to the MRS Manager GUI

MRS Manager provides a unified cluster management platform, facilitating rapid and easy O&M for clusters. For details about how to access MRS Manager, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Table 11-1 describes the functions of each operation entry.

Table 11-1 Functions of each entry on the operation bar

Parameter	Function
Dashboard	Displays the status of all services, main monitoring indicators of each service, and host status in charts, such as bar charts, line charts, and tables. You can customize a dashboard for the key monitoring indicators and drag it to any position on the interface. The system dashboard page supports automatic data update.
Services	Provides the service monitoring, operation, and configuration guidance, which helps you manage services in a unified manner.
Hosts	Provides guidance on how to monitor, operate, and configure hosts, helping you manage hosts in a unified manner.
Alarms	Supports alarm query and provides guidance on alarm handling, helping you identify and rectify product faults and potential risks in a timely manner to ensure normal system operation.
Audit	Allows authorized users to query and export audit logs, helping you to view all user activities and operations.
Tenant	Provides a unified tenant management platform.
System	Provides monitoring, alarm configuration management, and backup management.

Go to the **System** tab page, and switch to another function pages through shortcuts. See **Table 11-2**.

The following is an example of quick redirection through shortcuts:

Step 1 On MRS Manager, click **System**.

Step 2 On the **System** tab page, click a function link. The function page is displayed.

For example, in the **Backup and Restoration** area, click **Back Up Data**. The page for backing up data is displayed.

Step 3 Move the cursor to the left border of the browser window. The **System** black shortcut menu is displayed. After you move the cursor out of the menu, the menu is collapsed.

Step 4 In the shortcut menu that is displayed, you can click a function link to go to the corresponding function page.

For example, choose **Maintenance > Export Log**. The page for exporting logs is displayed.

----End

Table 11-2 Shortcut menus on the **System** tab page

Menu	Function Link
Backup and Restoration	Back Up Data
	Restore Data
Maintenance	Export Log
	Export Audit Log
	Check Health Status
Monitoring and Alarm	Configure Syslog
	Configure Alarm Threshold
	Configure SNMP
	Configure Monitoring Metric Dump
	Configure Resource Contribution Ranking
Permission	Manage User
	Manage User Group
	Manage Role
	Configure Password Policy
	Change OMS Database Password
Patch	Manage Patch

Reference

MapReduce Service (MRS) is a data analysis service. It is used to manage and analyze massive sets of data.

MRS uses MRS Manager to manage big data components, such as components in the Hadoop ecosystem. Therefore, some concepts on the MRS Console must be different from those on MRS Manager. For details, see **Table 11-3**.

Table 11-3 Difference Comparison

Concept	MRS	MRS Manager
MapReduce Service	Indicates the data analysis cloud service, called MRS. This service includes components such as Hive, Spark, Yarn, HDFS, and ZooKeeper.	Provides a unified management platform for big data components in tenant clusters.

11.2 Checking Running Tasks

Scenario

When you perform operations on MRS Manager to trigger a task, the task execution process and progress are displayed. After the task window is closed, you need to open the task window by using the task management function.

MRS Manager reserves 10 latest tasks by default, for example, restarting services, synchronizing service configurations, and performing health check.

Procedure

Step 1 On MRS Manager, click  to open the task list.

You can view the following information in the task list: **Name**, **Status**, **Progress**, **Start Time** and **End Time**.

Step 2 Click the target task name to view the detailed information about the running task.

----End

11.3 Monitoring Management

11.3.1 Dashboard

On MRS Manager, nodes in a cluster can be classified into management nodes, control nodes, and data nodes. The change trends of key host monitoring metrics on each type of node can be calculated and displayed as curve charts in reports based on the customized periods. If a host belongs to multiple node types, the metric statistics will be repeatedly collected.

This section provides overview of MRS clusters and describes how to view, customize, and export node monitoring metrics on MRS Manager.

Procedure

Step 1 Log in to MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 Choose **Dashboard** on MRS Manager.

Step 3 In **Period**, you can specify a period to view monitoring data. The options are as follows:

- Real time
- Last 3 hours
- Last 6 hours
- Last 24 hours

- Last week
- Last month
- Last 3 months
- Last 6 months
- Customize. If you select this option, you can customize the period for viewing monitoring data.

Step 4 Click **View** to view monitoring data in a period.

- You can view **Health Status** and **Roles** of each service on the **Service Summary** page of MRS Manager.
- Click  above the curve chart to view details about a metric.

Step 5 Customize a monitoring report.

1. Click **Customize** and select monitoring metrics to be displayed on MRS Manager.

   MRS Manager supports a maximum of 14 monitoring metrics, but at most 12 customized monitoring metrics can be displayed on the page.

   - Cluster Host Health Status
   - Cluster Network Read Speed Statistics
   - Host Network Read Speed Distribution
   - Host Network Write Speed Distribution
   - Cluster Disk Write Speed Statistics
   - Cluster Disk Usage Statistics
   - Cluster Disk Information
   - Host Disk Usage Statistics
   - Cluster Disk Read Speed Statistics
   - Cluster Memory Usage Statistics
   - Host Memory Usage Distribution
   - Cluster Network Write Speed Statistics
   - Host CPU Usage Distribution
   - Cluster CPU Usage Statistics
2. Click **OK** to save the selected monitoring metrics for display.

   NOTE

   Click **Clear** to cancel all the selected monitoring metrics in a batch.

Step 6 Set an automatic refresh interval or click  for an immediate refresh.

The following refresh interval options are supported:

- Refresh every 30 seconds
- Refresh every 60 seconds
- Stop refreshing

NOTE

If you select **Full Screen**, the **Dashboard** window will be maximized.

Step 7 Export a monitoring report.

1. Select a period. The options are as follows:
   - Real time
   - Last 3 hours
   - Last 6 hours
   - Last 24 hours
   - Last week
   - Last month
   - Last 3 months
   - Last 6 months
   - Customize. If you select this option, you can customize a time of period to export a report.
2. Click **Export**. MRS Manager will generate a report about the selected monitoring metrics in a specified time of period. Save the report.

   NOTE

   To view the curve charts of monitoring metrics in a specified period, click **View**.

----End

11.3.2 Managing Services and Monitoring Hosts

You can manage the following status and indicators of all services (including role instances) and hosts on the MRS Manager:

- Status information: includes operation, health, configuration, and role instance status.
- Metric information: includes key monitoring metrics for services.
- Metric export: allows you to export monitoring reports.

NOTE

Set an automatic refresh interval or click  for an immediate refresh.

The following refresh interval options are supported:

- Refresh every 30 seconds
- Refresh every 60 seconds
- Stop refreshing

Managing Service Monitoring

Step 1 On MRS Manager, click **Services**.

The service list includes **Service**, **Operating Status**, **Health Status**, **Configuration Status**, **Roles**, and **Operation** are displayed in the component list.

- **Table 11-4** describes the service operating status.

Table 11-4 Service operating status

Status	Description
Started	The service is started.
Stopped	The service is stopped.
Failed to start	Failed to start the role instance.
Failed to stop	Failed to stop the role instance.
Unknown	Indicates initial service status after the background system restarts.

- Table 11-5 describes the service health status.

Table 11-5 Service health status

Status	Description
Good	Indicates that all role instances in the service are running properly.
Bad	Indicates that the running status of at least one role instance is **Faulty** or the status of the service on which the current service depends is abnormal.
Unknown	Indicates that all role instances in the service are in the **Unknown** state.
Concerning	Indicates that the background system is restarting the service.
Partially Healthy	Indicates that the status of the service on which the service depends is abnormal, and APIs related to the abnormal service cannot be invoked by external systems.

- Table 11-6 describes the service health status.

Table 11-6 Service configuration status

Status	Description
Synchronized	The latest configuration takes effect.
Expired	The latest configuration does not take effect after the parameter modification. Related services need to be restarted.
Failed	The communication is incorrect or data cannot be read or written during the parameter configuration. Use **Synchronize Configuration** to rectify the fault.

Status	Description
Configuring	Parameters are being configured.
Unknown	Current configuration status cannot be obtained.

By default, the **Service** column is sorted in ascending order. You can click the icon next to **Service**, **Operating Status**, **Health Status**, or **Configuration Status** to change the sorting mode.

Step 2 Click a specified service in the list to view its status and metric information.

Step 3 Customize monitoring metrics and export customized monitoring information.

1. In the **Charts** area, click **Customize** to customize service monitoring metrics.
2. In **Period** area, select a time of period and click **View** to view the monitoring data within the time period.
3. Click **Export** to export the displayed metrics.

----End

Managing Role Instances

Step 1 On MRS Manager, click **Services** and click the target service name in the service list.

Step 2 Click **Instance** to view the role status.

The role instance list contains the **Role**, **Host Name**, **OM IP Address**, **Business IP Address**, **Rack**, **Operation Status**, **Health Status,** and **Configuration Status** of an instance.

- Table 11-7 shows the configuration status of a role instance.

Table 11-7 Role instance status

Status	Description
Started	The role instance has been started.
Stopped	The role instance has been stopped.
Failed to start	Failed to start the role instance.
Failed to stop	Failed to stop the role instance.
Decommissioning	The role instance is being decommissioned.
Decommissioned	The role instance has been decommissioned.
Recommissioning	The role instance is being recommissioned.
Unknown	Indicates initial role instance status after the background system restarts.

- Table 11-8 shows the health status of a role instance.

  Table 11-8 Role instance health status

Status	Description
Good	The role instance is running properly.
Bad	The role instance is abnormal. For example, the port cannot be accessed if PID does not exist.
Unknown	The host where a role instance resides does not connect to the background system.
Concerning	The background system is restarting a role instance.

- Table 11-9 shows the configuration status of a role instance.

  Table 11-9 Role instance configuration status

Status	Description
Synchronized	The latest configuration takes effect.
Expired	The latest configuration does not take effect after the parameter modification. Related services need to be restarted.
Failed	The communication is incorrect or data cannot be read or written during the parameter configuration. Use **Synchronize Configuration** to rectify the fault.
Configuring	Parameters are being configured.
Unknown	Current configuration status cannot be obtained.

By default, the **Role** column is sorted in ascending order. You can click the sorting icon next to **Role**, **Host Name**, **OM IP Address**, **Business IP Address**, **Rack**, **Operating Status**, **Health Status**, or **Configuration Status** to change the sorting mode.

You can filter out all instances of the same role in the **Role** column.

You can set search criteria in the role search area by clicking **Advanced Search**, and click **Search** to view specified role information. Click **Reset** to clear the search criteria. Fuzzy search is supported.

Step 3 Click the target role instance to view its status and metric information.

Step 4 Customize monitoring metrics and export customized monitoring information.

1. In the **Charts** area, click **Customize** to customize service monitoring metrics.
2. In **Period** area, select a time of period and click **View** to view the monitoring data within the time period.
3. Click **Export** to export the displayed metrics.

----End

Managing Hosts

Step 1 On MRS Manager, click **Hosts** to view the status of all hosts.

The host list contains the host name, management IP address, service IP address, rack, network speed, operating status, health status, disk usage, memory usage, and CPU usage.

- **Table 11-10** shows the host operating status.

  Table 11-10 Host operating status

Status	Description
Normal	The host and service roles on the host are running properly.
Isolated	The host is isolated, and the service roles on the host stop running.

- **Table 11-11** describes the host health status.

  Table 11-11 Host health status

Status	Description
Good	The host can properly send heartbeats.
Bad	The host fails to send heartbeats due to timeout.
Unknown	The host initial status is unknown during the operation of adding or deleting a host.

By default, the **Host Name** column is sorted by host name in ascending order. You can click the sorting icon next to **Host Name**, **OM IP Address**, **Business IP Address**, **Rack**, **Network Speed**, **Operating Status**, **Health Status**, **Disk Usage**, **Memory Usage**, or **CPU Usage** to change the sorting mode.

You can set search criteria in the role search area by clicking **Advanced Search**, and click **Search** to view specified role information. Click **Reset** to clear the search criteria. Fuzzy search is supported.

Step 2 Click the target host in the host list to view its status and metric information.

Step 3 Customize monitoring metrics and export customized monitoring information.

1. In the **Charts** area, click **Customize** to customize service monitoring metrics.
2. In **Period** area, select a time of period and click **View** to view the monitoring data within the time period.
3. Click **Export** to export the displayed metrics.

----End

11.3.3 Managing Resource Distribution

On MRS Manager, you can query the top value curves, bottom value curves, or average data curves of key service and host monitoring metrics, that is, the resource distribution information. MRS Manager allows you to view the monitoring data of the last hour.

You can also modify the resource distribution on MRS Manager to display both the top and bottom value curves in service and host resource distribution figures.

Resource distribution of some monitoring metrics is not recorded.

Procedure

- View the resource distribution of service monitoring metrics.
  a. On MRS Manager, click **Services**.
  b. Select the target service from the service list.
  c. Click **Resource Distribution**.
     Select key metrics of the service from **Metric**. MRS Manager displays the resource distribution of the metrics in the last hour.
- View the resource distribution of host monitoring metrics.
  a. Click **Hosts**.
  b. Click the name of the specified host in the host list.
  c. Click **Resource Distribution**.
     Select key metrics of the host from **Metrics**. MRS Manager displays the resource distribution of the metrics in the last hour.
- Configure resource distribution.
  a. On MRS Manager, click **System**.
  b. In **Configuration**, click **Configure Resource Contribution Ranking** under **Monitoring and Alarm**.
  c. Change the number of resources to be displayed.
     - Set **Number of Top Resources** to the number of top values.
     - Set **Number of Bottom Resources** to the number of bottom values.

     NOTE

     The sum of the maximum value and minimum value of resource distribution cannot be greater than 5.
  d. Click **OK** to save the configurations.
     The message "Number of top and bottom resources saved successfully" is displayed in the upper right corner of the page.

11.3.4 Configuring Monitoring Metric Dumping

You can configure interconnection parameters on MRS Manager to save monitoring metric data to a specified FTP server using the FTP or SFTP protocol. In this way, MRS clusters can interconnect with third-party systems. The FTP protocol does not encrypt data, which brings potential security risks. Therefore, the SFTP protocol is recommended.

MRS Manager supports the collection of all the monitoring metric data in the managed clusters. The collection period is 30 seconds, 60 seconds, or 300 seconds. The monitoring metric data is stored to different monitoring files on the FTP server by collection period. The monitoring file naming rule is in the "*Cluster name*_**metric**_*Monitoring metric data collection period*_*File saving time*.**log**" format.

Prerequisites

The ECS corresponding to the dump server must be in the same VPC as the Master node of the MRS cluster, and the Master node can access the IP address and specified port of the dump server. The FTP service on the dump server is running properly.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In **Configuration**, click **Configure Monitoring Metric Dump** under **Monitoring and Alarm**.

Step 3 Table 11-12 describes dump parameters.

Table 11-12 Dump parameters

Parameter	Description
FTP IP Address	Mandatory. This parameter specifies the FTP server for storing monitoring files after the monitoring indicator data is interconnected.
FTP Port	Mandatory. This parameter specifies the port connected to the FTP server.
FTP Username	Mandatory. This parameter specifies the username for logging in to the FTP server.
FTP Password	Mandatory. This parameter specifies the password for logging in to the FTP server.
Save Path	Mandatory. This parameter specifies the path for storing monitoring files on the FTP server.
Dump Interval (s)	Mandatory. This parameter specifies the interval at which monitoring files are periodically stored on the FTP server, in seconds.
Dump Mode	Mandatory. This parameter specifies the protocol used for sending monitoring files. This parameter is mandatory. The options are **FTP** and **SFTP**.
SFTP Public Key	Optional. This parameter specifies the public key of the FTP server and is valid only when **Dump Mode** is set to **SFTP**. You are advised to configure a public key. Otherwise, security risks may arise.

Step 4 Click **OK** to complete the settings.

----End

11.4 Alarm Management

11.4.1 Viewing and Manually Clearing an Alarm

Scenario

You can view and clear alarms on MRS Manager.

Generally, the system automatically clears an alarm when the fault is rectified. If the fault has been rectified and the alarm cannot be automatically cleared, you can manually clear the alarm.

You can view the latest 100,000 alarms (including uncleared, manually cleared, and automatically cleared alarms) on MRS Manager. If the number of cleared alarms exceeds 100,000 and is about to reach 110,000, the system automatically dumps the earliest 10,000 cleared alarms to **${BIGDATA_HOME}/OMSV100R001C00x8664/workspace/data** on the active management node. A directory is automatically generated when alarms are dumped for the first time.

NOTE

Set an automatic refresh interval or click  for an immediate refresh.

The following refresh interval options are supported:

- Refresh every 30 seconds
- Refresh every 60 seconds
- Stop refreshing

Procedure

Step 1 On MRS Manager, click **Alarms** to view the alarm information in the alarm list.

- By default, the alarm list page displays the latest 10 alarms.
- By default, alarms are displayed in descending order by **Generated**. You can click **Alarm ID**, **Alarm Name**, **Severity**, **Generated**, **Location**, **Operation** to change the display mode.
- You can filter all alarms of the same severity in **Severity**, including cleared and uncleared alarms.
- You can click  ,  , , or  to filter out **Critical**, **Major**, **Minor**, or **Warning** alarms.

Step 2 Click **Advanced Search**. In the displayed alarm search area, set search criteria and click **Search** to view the information about specified alarms. Click **Reset** to clear the search criteria.

NOTE

You can set the **Start Time** and **End Time** to specify the time range. You can search for alarms generated within the time range.

Handle the alarm by referring to **Alarm Reference**. If the alarms in some scenarios are generated due to other cloud services that MRS depends on, you need to contact maintenance personnel of the corresponding cloud services.

Step 3 If the alarm needs to be manually cleared after errors are rectified, click **Clear Alarm**.

NOTE

If multiple alarms have been handled, you can select one or more alarms to be cleared and click **Clear Alarm** to clear the alarms in batches. A maximum of 300 alarms can be cleared in each batch.

----End

11.4.2 Configuring an Alarm Threshold

Scenario

You can configure an alarm threshold to learn the metric health status. After **Send Alarm** is selected, the system sends an alarm message when the monitored data reaches the alarm threshold. You can view the alarm information in **Alarms**.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In **Configuration**, click **Configure Alarm Threshold** under **Monitoring and Alarm**, select monitoring metrics as planned, and set their baselines.

Step 3 Click a metric, for example, **CPU Usage**, and click **Create Rule**.

Step 4 Set the monitoring metric rule parameters on the displayed configuration page.

Table 11-13 Monitoring metric rule parameters

Parameter	Value	Description
Rule Name	CPU_MAX (example value)	Specifies the rule name.
Reference Date	2014/11/06 (example)	Specifies the date on which the reference indicator history is generated.

Parameter	Value	Description
Threshold Type	• Max. value • Min. value	Specifies the maximum or minimum value of a metric. If this parameter is set to **Max. Value**, the system generates an alarm when the actual value of the metric is greater than the threshold. If this parameter is set to **Min. Value**, the system generates an alarm when the actual value of the metric is smaller than the threshold.
Alarm Severity	• Critical • Major • Minor • Suggestion	Alarm Severity
Time Range	From 00:00 to 23:59 (example)	Specifies the period in which the rule takes effect.
Threshold	80 (example)	Specifies the threshold of the rule monitoring metrics.
Date	• Workday • Weekend • Other	Specifies the type of date when the rule takes effect.
Add Date	11/06 (example)	This parameter is valid only when **Date** is set to **Other**. You can select multiple dates.

Step 5 Click **OK**. A message is displayed in the upper right corner of the page, indicating that the template is saved successfully.

Send alarm is selected by default. MRS Manager checks whether the value of each monitored metric reaches the threshold. If the number of consecutive check times is equal to the value of **Trigger Count**, and the threshold is not reached in these checks, the system sends an alarm. The value can be customized. **Check Period (s)** indicates the interval at which MRS Manager checks monitoring metrics.

Step 6 Locate the row that contains the newly added rule, and click **Apply** in the **Operation** column. A message is displayed in the upper right corner, indicating that the rule *xx* is successfully added. Click **Cancel** in the **Operation** column. A

message is displayed in the upper right corner, indicating that the rule *xx* is successfully canceled.

----End

11.4.3 Configuring Syslog Northbound Interface Parameters

Scenario

You can configure the northbound interface so that alarms generated on MRS Manager can be reported to your monitoring O&M system using Syslog.

NOTICE

If the Syslog protocol is not encrypted, data may be stolen.

Prerequisites

The ECS corresponding to the server must be in the same VPC as the Master node of the MRS cluster, and the Master node can access the IP address and specified port of the server.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In **Configuration**, click **Configure Syslog** under **Monitoring and Alarm**.

The **Syslog Service** is disabled by default. Click the switch to enable the Syslog service.

Step 3 Set the interconnection parameters listed in **Table 11-14**.

Table 11-14 Syslog parameters

Area	Parameter	Description
Syslog Protocol	Service IP Address	Specifies the IP address of the interconnection server.
	Server Port	Specifies the port number for interconnection.
	Protocol	Specifies the protocol type. The options are as follows: • **TCP** • **UDP**

Area	Parameter	Description
	Severity	Specifies the severity of the reported message. The options are as follows: • **Informational** • **Emergency** • **Alert** • **Critical** • **Error** • **Warning** • **Notice** • **Debug**
	Facility	Specifies the module where the log is generated.
	Identifier	Specifies the product ID. The default value is **MRS Manager**.
Report Message	Report Format	Specifies the message format of the alarm report. For details, see help information on the web page.
	Alarm Status	Specifies the type of the alarm to be reported. • **Fault**: indicates that the Syslog alarm message is reported when MRS Manager generates an alarm. • **Clear**: indicates that a Syslog alarm message is reported when an alarm on MRS Manager is cleared. • **Event**: indicates that the Syslog alarm message is reported when MRS Manager generates an event.

Area	Parameter	Description
	Report Alarm Severity	Specifies the level of the alarm to be reported. The value can be **Suggestion**, **Minor**, **Major**, and **Critical**.
Uncleared Alarm Reporting	Periodic Uncleared Alarm Report	Specifies whether uncleared alarms are reported periodically. By default, the switch of **Periodic Uncleared Alarm Reporting** is disabled. You can click the switch to enable it.
	Report Interval (min)	Specifies the interval for periodically reporting uncleared alarms to the remote Syslog service. This parameter is valid only when **Periodic Uncleared Alarm Reporting** switch is enabled. The unit is minute. The default value is **15**. The value ranges from 5 minutes to one day (1,440 minutes).
Heartbeat Settings	Heartbeat Report	Specifies whether to periodically report Syslog heartbeat messages. By default, the switch of **Periodic Uncleared Alarm Reporting** is disabled. You can click the switch to enable it.
	Heartbeat Period (min)	Specifies the interval for periodically reporting heartbeat messages. This parameter is valid only when **Heartbeat Report** switch is enabled. The unit is minute. The default value is **15**. The value ranges from 1 to 60.

Area	Parameter	Description
	Heartbeat Packet	Specifies the content of the reported heartbeat message. This parameter is enabled when **Heartbeat Report** is enabled. The value can contain a maximum of 256 characters, including digits, letters, underscores (_), vertical bars (|), colons (:), spaces, commas (,), and periods (.).

NOTE

After the periodic heartbeat packet function is enabled, packets may be interrupted during automatic recovery of some cluster error tolerance (for example, active/standby management node switchover). In this case, wait for automatic recovery.

Step 4 Click **OK** to complete the settings.

----End

11.4.4 Configuring SNMP Northbound Interface Parameters

Scenario

You can configure the northbound interface so that alarms and monitoring metrics on MRS Manager can be integrated to the network management platform using SNMP.

Prerequisites

The ECS corresponding to the server must be in the same VPC as the Master node of the MRS cluster, and the Master node can access the IP address and specified port of the server.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In **Configuration**, click **Configure SNMP** under **Monitoring and Alarm**.

The **SNMP Service** is disabled by default. Click the switch to enable the SNMP service.

Step 3 Set the interconnection parameters listed in **Table 11-15**.

Table 11-15 Syslog parameters

Parameter	Description
Version	Specifies the version of the SNMP, which can be: • v2c: an earlier version with low security • v3: the latest version of SNMP with higher security than SNMPv2c The SNMP v3 version is recommended.
Local Port	Specifies the local port. The default value is **20000**. The value ranges from **1025** to **65535**.
Read Community Name	Specifies the read-only community name. This parameter is valid only when **Version** is set to **v2c**.
Write Community Name	Specifies the write community name. This parameter is valid only when **Version** is set to **v2c**.
Security Username	Specifies the SNMP security username. This parameter is valid only when **Version** is set to **v3**.
Authentication Protocol	Specifies the authentication protocol. You are advised to set this parameter to set this parameter to **SHA**. This parameter is valid only when **Version** is set to **v3**.
Authentication Password	Specifies the authentication key. This parameter is valid only when **Version** is set to **v3**.
Confirm Password	Used to confirm the authentication key. This parameter is valid only when **Version** is set to **v3**.
Encryption Protocol	Specifies the encryption protocol. You are advised to set this parameter to **AES256**. This parameter is valid only when **Version** is set to **v3**.
Encryption Password	Specifies the encryption key. This parameter is valid only when **Version** is set to **v3**.
Confirm Password	Used to confirm the encryption key. This parameter is valid only when **Version** is set to **v3**.

NOTE

- The **Authentication Password** and **Encryption Password** must contain 8 to 16 characters, including at least three types of the following characters: uppercase letters, lowercase letters, digits, and special characters. The two passwords must be different. The two passwords cannot be the same as the security username or the reverse of the security username.
- For security purposes, periodically change the authentication password and encryption password when the SNMP protocol is used.
- If SNMPv3 is used, a security user will be locked after five consecutive authentication failures within 5 minutes. The user will be automatically unlocked 5 minutes later.

Step 4 Click **Create Trap Target** in the **Trap Target** area. In the displayed dialog box, set the following parameters:

- **Target Symbol** specifies the trap target ID, which is the ID of the NMS or host that receives traps. The value consists of 1 to 255 characters, including letters or digits.
- **Target IP Address** specifies the IP address of the target trap. IP addresses of class A, B, and C can be used to communicate with the IP address of the management plane of the management node.
- **Target Port** specifies the port receiving traps. The port number must be consistent with the peer end and ranges from 0 to 65535.
- **Trap Community Name** is valid only when **Version** is set to **v2c**.

Click **OK**. The **Create Trap Target** dialog box is closed.

Step 5 Click **OK** to complete the settings.

----End

11.5 Object Management

11.5.1 Managing Objects

MRS contains different types of basic objects as described in **Table 11-16**.

Table 11-16 MRS basic object overview

Object	Description	Example
Service	Function set that can complete specific business.	KrbServer service and LdapServer service
Service instance	Specific instance of a service, usually called service.	KrbServer service
Service role	Function entity that forms a complete service, usually called role.	KrbServer is composed of the KerberosAdmin role and KerberosServer role.
Role instance	Specific instance of a service role running on a host.	KerberosAdmin that is running on Host2 and KerberosServer that is running on Host3
Host	An ECS running Linux OS.	Host1 to Host5
Rack	Physical entity that contains multiple hosts connecting to the same switch.	Rack1 contains Host1 to Host5.
Cluster	Logical entity that consists of multiple hosts and provides various services.	Cluster names **Cluster1** consists of five hosts (Host1 to Host5) and provides services such as KrbServer and LdapServer.

11.5.2 Viewing Configurations

On MRS Manager, users can view the configurations of services (including roles) and role instances.

Procedure

- Query service configurations.
  a. On MRS Manager page, click **Services**.
  b. Select the target service from the service list.
  c. Click **Service Configuration**.
  d. Set **Type** to **All**. All configuration parameters of the service are displayed in the navigation tree. The root nodes from top down in the navigation tree represent the service names and role names.
  e. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

     The parameters under the service nodes and role nodes are service configuration parameters and role configuration parameters respectively.
  f. In the **Non-default** parameter, select **Non-default**. The parameters whose values are not default values will be displayed.
- Query role instance configurations.
  a. On MRS Manager page, click **Services**.
  b. Select the target service from the service list.
  c. Click the **Instances** tab.
  d. Click the target role instance from the role instance list.
  e. Click **Instance Configuration**.
  f. Set **Type** to **All**. The navigation tree of all configuration parameters of the role instance is displayed.
  g. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.
  h. In the **Non-default** parameter, select **Non-default**. The parameters whose values are not default values will be displayed.

11.5.3 Managing Services

You can perform the following operations on MRS Manager:

- Start the service in the **Stopped**, **Stop Failed**, or **Start Failed** state to use the service.
- Stop the services or stop abnormal services.
- Restart abnormal services or configure expired services to restore or enable the services.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 Locate the row that contains the target service, **Start**, **Stop**, or **Restart** to start, stop, or restart the service.

Services are interrelated. If a service is started, stopped, and restarted, services dependent on it will be affected.

The services will be affected in the following ways:

- If a service is to be started, the lower-layer services dependent on it must be started first.
- If a service is stopped, the upper-layer services dependent on it are unavailable.
- If a service is restarted, the running upper-layer services dependent on it must be restarted.

----End

11.5.4 Configuring Service Parameters

On MRS Manager, you can view and modify the default service configurations based on site requirements and export or import the configurations.

Impact on the System

- You need to download and update the client configuration files after configuring HBase, HDFS, Hive, Spark, Yarn, and MapReduce service properties.
- The parameters of DBService cannot be modified when only one DBService role instance exists in the cluster.

Procedure

- Modify a service.
  a. Click **Services**.
  b. Select the target service from the service list.
  c. Click **Service Configuration**.
  d. Set **Type** to **All**. All configuration parameters of the service are displayed in the navigation tree. The root nodes from top down in the navigation tree represent the service names and role names.
  e. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

     If you want to cancel the modification of a parameter value, click  to restore it.

NOTE

You can also use host groups to change role instance configurations in batches. Select a role name from the **Role** drop-down list and choose **< Select Host >** in the **Host** drop-down list. Enter a name in the **Host Group Name** text box, select the hosts to be modified from the **Host** list, add them to the **Selected hosts** area, and click **OK**. The added host group can be selected from **Host** and is only valid on the current page. The page cannot be saved after being refreshed.

f. Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the services.

After **Operation successful.** is displayed, click **Finish**. The service is started successfully.

NOTE

To update the queue configuration of the Yarn service without restarting service, choose **More** > **Refresh Queue** to update the queue for the configuration to take effect.

- Export service configuration parameters.
  a. Click **Services**.
  b. Select a service.
  c. Click **Service Configuration**.
  d. Click **Export Service Configuration**. Select a path for saving the configuration files.
- Import service configuration parameters.
  a. Click **Services**.
  b. Select a service.
  c. Click **Service Configuration**.
  d. Click **Import Service Configuration**.
  e. Select the target configuration file.
  f. Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK**.

After **Operation successful.** is displayed, click **Finish**. The service is started successfully.

11.5.5 Configuring Customized Service Parameters

Each component of MRS supports all open-source parameters. You can modify some parameters for key application scenarios on MRS Manager. Some component clients may not include all parameters with open-source features. For component parameters that cannot be directly modified on Manager, users can add new parameters for components by using the configuration customization function on Manager. Newly added parameters are saved in component configuration files and take effect after restart.

Impact on the System

- After the service attributes are configured, the service needs to be restarted and cannot be accessed.

- You need to download and update the client configuration files after configuring HBase, HDFS, Hive, Spark, Yarn, and MapReduce service properties.

Prerequisites

You have understood the meanings of parameters to be added, configuration files that have taken effect, and the impact on components.

Procedure

Step 1 On MRS Manager, click **Services**.

Step 2 Select the target service from the service list.

Step 3 Click **Service Configuration**.

Step 4 Set **Type** to **All**.

Step 5 In the navigation tree, select **Customization**. The customized parameters of the current component are displayed on Manager.

The configuration files that save the newly added customized parameters are displayed in the **Parameter File** column. Different configuration files may have same open source parameters. After the parameters in different files are set to different values, whether the configuration takes effect depends on the loading sequence of the configuration files by components. You can customize parameters for services and roles as required. Adding customized parameters for a single role instance is not supported.

Step 6 Based on the configuration files and parameter functions, locate the row where a specified parameter resides, enter the parameter name supported by the component in the **Name** column and enter the parameter value in the **Value** column.

- You can click  or  to add or delete a user-defined parameter. You can delete a customized parameter only after you click  for the first time.
- If you want to cancel the modification of a parameter value, click  to restore it.

Step 7 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the services.

After **Operation successful.** is displayed, click **Finish**. The service is started successfully.

----End

Task Example

Configuring Customized Hive Parameters

Hive depends on HDFS. By default, Hive accesses the HDFS client. The configuration parameters to take effect are controlled by HDFS in a unified manner. For example, the HDFS parameter **ipc.client.rpc.timeout** affects the RPC timeout period for all clients to connect to the HDFS server. If you need to modify

the timeout period for Hive to connect to HDFS, you can use the configuration customization function. After this parameter is added to the **core-site.xml** file of Hive, this parameter can be identified by the Hive service and its configuration overwrites the parameter configuration in HDFS.

Step 1 On MRS Manager, choose **Services** > **Hive** > **Service Configuration**.

Step 2 Set **Type** to **All**.

Step 3 In the navigation tree on the left, select **Customization** for the Hive service. The system displays the customized service parameters supported by Hive.

Step 4 In **core-site.xml**, locate the row that contains the **core.site.customized.configs** parameter, enter **ipc.client.rpc.timeout** in the **Name** column, and enter a new value in the **Value** column, for example, **150000**. The unit is millisecond.

Step 5 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the service.

After **Operation successful.** is displayed, click **Finish**. The service is started successfully.

----End

11.5.6 Synchronizing Service Configurations

Scenario

If **Configuration Status** of a service is **Expired** or **Failed**, synchronize configurations for the cluster or service to restore its configuration status. If all services in the cluster are in the **Failed** state, synchronize the cluster configuration with the background configuration.

Impact on the System

After synchronizing service configurations, you need to restart the services whose configurations have expired. These services are unavailable during restart.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 Select the target service from the service list.

Step 3 In the upper part of the service status and metric information, choose **More** > **Synchronize Configuration**.

Step 4 In the displayed dialog box, select **Restart services and instances whose configuration have expired.** and click **OK** to restart the service whose configuration has expired.

When **Operation successful** is displayed, click **Finish**. The service is started successfully.

----End

11.5.7 Managing Role Instances

Scenario

You can start a role instance that is in the **Stopped**, **Failed to stop** or **Failed to start** status, stop an unused or abnormal role instance or restart an abnormal role instance to recover its functions.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 Select the target service from the service list.

Step 3 Click the **Instances** tab.

Step 4 Select the check box on the left of the target role instance.

Step 5 Choose **More** > **Start Instance**, **Stop Instance**, or **Restart Instance** accordingly.

----End

11.5.8 Configuring Role Instance Parameters

Scenario

You can view and modify default role instance configurations on MRS Manager based on site requirements. The configurations can be imported and exported.

Impact on the System

You need to download and update the client configuration files after configuring HBase, HDFS, Hive, Spark, Yarn, and MapReduce service properties.

Procedure

- Modifying role instance configurations
  a. Click **Services**.
  b. Select the target service from the service list.
  c. Click the **Instances** tab.
  d. Click the target role instance from the role instance list.
  e. Click **Instance Configuration**.
  f. Set **Type** to **All**. The navigation tree of all configuration parameters of the role instance is displayed.
  g. In the navigation tree, select a specified parameter and change its value. You can also enter the parameter name in the **Search** box to search for the parameter and view the result.

     If you want to cancel the modification of a parameter value, click  to restore it.
  h. Click **Save Configuration**, select **Restart the role instance**, and click **OK** to restart the role instance.

After **Operation successful.** is displayed, click **Finish**. The role instance is started successfully.

- Exporting Configuration Parameters of a Role Instance
  a. Click **Services**.
  b. Select a service.
  c. Select a role instance or click the **Instances** tab.
  d. Select a role instance on a specified host.
  e. Click **Instance Configuration**.
  f. Click **Export Instance Configuration** to export the configuration data of a specified role instance, and choose a path for saving the configuration file.
- Import configuration data of a role instance.
  a. Click **Services**.
  b. Select a service.
  c. Select a role instance or click the **Instances** tab.
  d. Select a role instance on a specified host.
  e. Click **Instance Configuration**.
  f. Click **Import Instance Configuration** to import the configuration data of the specified role instance.
  g. Click **Save Configuration** and select **Restart the role instance**. Click **OK**.
  After **Operation successful.** is displayed, click **Finish**. The role instance is started successfully.

11.5.9 Synchronizing Role Instance Configuration

Scenario

When **Configuration Status** of a role instance is **Expired** or **Failed**, you can synchronize the configuration data of the role instance with the background configuration.

Impact on the System

After synchronizing a role instance configuration, you need to restart the role instance whose configuration has expired. The role instance is unavailable during restart.

Procedure

Step 1 On MRS Manager, click **Services** and select a service name.

Step 2 Click the **Instances** tab.

Step 3 Click the target role instance from the role instance list.

Step 4 Choose **More** > **Synchronize Configuration** above the role instance status and indicator information.

Step 5 In the displayed dialog box, select **Restart services and instances whose configuration have expired**, and click **OK** to restart the role instance.

After **Operation successful** is displayed, click **Finish**. The role instance is started successfully.

----End

11.5.10 Decommissioning and Recommissioning a Role Instance

Scenario

If a Core or Task node is faulty, the cluster status may be displayed as **Abnormal**. In an MRS cluster, data can be stored on different Core nodes. Users can decommission the specified role instance on MRS Manager to stop the role instance from providing services. After fault rectification, you can recommission the role instance.

The following role instances can be decommissioned and recommissioned.

- DataNode role instance on HDFS
- NodeManager role instance on Yarn
- RegionServer role instance on HBase
- Broker role instance on Kafka

Restrictions:

- If the number of the DataNodes is less than or equal to that of HDFS copies, decommissioning cannot be performed. If the number of HDFS copies is three and the number of DataNodes is less than four in the system, decommissioning cannot be performed. In this case, an error will be reported and the decommissioning will be stopped 30 minutes after the decommissioning attempt is performed on Manager.
- If the number of Kafka Broker instances is less than or equal to that of copies, decommissioning cannot be performed. For example, if the number of Kafka copies is two and the number of nodes is less than three in the system, decommissioning cannot be performed. Instance decommissioning will fail on Manager and exit.
- If a role instance is out of service, you must recommission the instance to start it before using it again.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 Click a service in the service list.

Step 3 Click the **Instances** tab.

Step 4 Select an instance.

Step 5 Choose **More** > **Decommission** or **Recommission** to perform the corresponding operation.

NOTE

During the instance decommissioning, if the service corresponding to the instance is restarted in the cluster using another browser, MRS Manager displays a message indicating that the instance decommissioning is stopped, but the **Operating Status** of the instance is displayed as **Started**. In this case, the instance has been decommissioned on the background. You need to decommission the instance again to synchronize the operating status.

----End

11.5.11 Managing a Host

Scenario

When a host is abnormal or faulty, you need to stop all roles of the host on MRS Manager to check the host. After the host fault is rectified, start all roles running on the host to recover host services.

Procedure

Step 1 Click **Hosts**.

Step 2 Select the check box of the target host.

Step 3 Choose **More** > **Start All Roles** or **Stop All Roles** accordingly.

----End

11.5.12 Isolating a Host

Scenario

If a host is found to be abnormal or faulty, affecting cluster performance or preventing services from being provided, you can temporarily exclude that host from the available nodes in the cluster. In this way, the client can access other available nodes. In scenarios where patches are to be installed in a cluster, you can also exclude a specified node from patch installation.

Users can isolate a host manually on MRS Manager based on the actual service requirements or O&M plan. Only non-management nodes can be isolated.

Impact on the System

- After a host is isolated, all role instances on the host will be stopped. You cannot start, stop, or configure the host and any instances on the host.
- After a host is isolated, statistics about the monitoring status and indicator data of the host hardware and instances on the host cannot be collected or displayed.

Procedure

Step 1 On MRS Manager, click **Hosts**.

Step 2 Select the check box of the host to be isolated.

Step 3 Choose **More** > **Isolate Host**,

Step 4 and click **OK** in the displayed dialog box.

After **Operation successful.** is displayed, click **Finish**. The host is isolated successfully, and the value of **Operating Status** becomes **Isolated**.

NOTE

For isolated hosts, you can cancel the isolation and add them to the cluster again. For details, see **Canceling Host Isolation**.

----End

11.5.13 Canceling Host Isolation

Scenario

After the exception or fault of a host is handled, you must cancel the isolation of the host for proper usage.

Users can cancel the isolation of a host on MRS Manager.

Prerequisites

- The host is in the **Isolated** state.
- The exception or fault of the host has been rectified.

Procedure

Step 1 On MRS Manager, click **Hosts**.

Step 2 Select the check box of the host to be de-isolated.

Step 3 Choose **More** > **Cancel Host Isolation**,

Step 4 and click **OK** in the displayed dialog box.

After **Operation successful.** is displayed, click **Finish**. The host is de-isolated successfully, and the value of **Operating Status** becomes **Normal**.

Step 5 Click the name of the de-isolated host to show its status, and click **Start All Roles**.

----End

11.5.14 Starting or Stopping a Cluster

Scenario

A cluster is a collection of service components. You can start or stop all services in a cluster.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 In the upper part of the service list, choose **More** > **Start Cluster** or **Stop Cluster** accordingly.

----End

11.5.15 Synchronizing Cluster Configurations

Scenario

If **Configuration Status** of all services or some services is **Expired** or **Failed**, synchronize configuration for the cluster or service to restore its configuration status.

- If all services in the cluster are in the **Failed** state, synchronize the cluster configuration with the background configuration.
- If all services in the cluster are in the **Failed** state, synchronize the service configuration with the background configuration.

Impact on the System

After synchronizing cluster configurations, you need to restart the services whose configurations have expired. These services are unavailable during restart.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 In the upper part of the service list, choose **More** > **Synchronize Configuration**.

Step 3 In the displayed dialog box, select **Restart services and instances whose configuration have expired**, and click **OK** to restart the service whose configuration has expired.

When **Operation successful.** is displayed, click **Finish**. The service is started successfully.

----End

11.5.16 Exporting Configuration Data of a Cluster

Scenario

You can export all configuration data of a cluster on MRS Manager to meet site requirements. The exported configuration data is used to rapidly update service configuration.

Procedure

Step 1 On MRS Manager page, click **Services**.

Step 2 Choose **More** > **Export Cluster Configuration**.

The exported file is used to update service configurations. For details, see **Import service configuration parameters** in **Configuring Service Parameters**.

----End

11.6 Log Management

11.6.1 About Logs

Log Description

MRS cluster logs are stored in the **/var/log/Bigdata** directory. The following table lists the log types.

Table 11-17 Log types

Type	Description
Installation log	Installation logs record information about FusionInsight Manager, cluster, and service installation to help users locate installation errors.
Run logs	Run logs record the running track information, debugging information, status changes, potential problems, and error information generated during the running of services.
Audit logs	Audit logs record information about users' activities and operation instructions, which can be used to locate fault causes in security events and determine who are responsible for these faults.

The following table lists the MRS log directories.

Table 11-18 Log directories

File Directory	Log Content
/var/log/Bigdata/audit	Component audit log.
/var/log/Bigdata/controller	Log collecting script log. Controller process log. Controller monitoring log.
/var/log/Bigdata/dbservice	DBService log.
/var/log/Bigdata/flume	Flume log.
/var/log/Bigdata/hbase	HBase log.
/var/log/Bigdata/hdfs	HDFS log.
/var/log/Bigdata/hive	Hive log.

File Directory	Log Content
/var/log/Bigdata/httpd	HTTPD log.
/var/log/Bigdata/hue	Hue log.
/var/log/Bigdata/kerberos	Kerberos log.
/var/log/Bigdata/ldapclient	LDAP client log.
/var/log/Bigdata/ldapserver	LDAP server log.
/var/log/Bigdata/loader	Loader log.
/var/log/Bigdata/logman	logman script log management log.
/var/log/Bigdata/mapreduce	MapReduce log.
/var/log/Bigdata/nodeagent	NodeAgent log.
/var/log/Bigdata/okerberos	OMS Kerberos log.
/var/log/Bigdata/oldapserver	OMS LDAP log.
/var/log/Bigdata/omm	**oms**: complex event processing log, alarm service log, HA log, authentication and authorization management log, and monitoring service run log of the omm server. **oma**: installation log and run log of the omm agent. **core**: dump log generated when the omm agent and the HA process are suspended.
/var/log/Bigdata/spark	Spark log.
/var/log/Bigdata/sudo	Log generated when the **sudo** command is executed by user **omm**.
/var/log/Bigdata/timestamp	Time synchronization management log.
/var/log/Bigdata/tomcat	Tomcat log.
/var/log/Bigdata/yarn	Yarn log.
/var/log/Bigdata/zookeeper	ZooKeeper log.
/var/log/Bigdata/kafka	Kafka log.
/var/log/Bigdata/storm	Storm log.
/var/log/Bigdata/patch	Patch log.

Run logs

Table 11-19 describes the running information recorded in run logs.

Table 11-19 Running information

Run Log	Description
Installation preparation log	Records information about preparations for the installation, such as the detection, configuration, and feedback operation information.
Process startup log	Records information about the commands executed during the process startup.
Process startup exception log	Records information about exceptions during process startup, such as dependent service errors and insufficient resources.
Process run log	Records information about the process running track information and debugging information, such as function entries and exits as well as cross-module interface messages.
Process running exception log	Records errors that cause process running errors, for example, the empty input objects or encoding or decoding failure.
Process running environment log	Records information about the process running environment, such as resource status and environment variables.
Script logs	Records information about the script execution process.
Resource reclamation log	Records information about the resource reclaiming process.
Uninstallation clearing logs	Records information about operations performed during service uninstallation, such as directory deletion and execution time

Audit logs

Audit information recorded in audit logs includes FusionInsight Manager audit information and component audit information.

Table 11-20 Audit information of FusionInsight Manager

Audit Log	Operation Type	Operation
Manager audit log	User management	Creating a user Modifying a user Deleting a user Creating a user group Modifying a user group Deleting a user group Adding a role Modifying a role Deleting a role Changing a password policy Changing a password Resetting a password User login User logout Unlocking the screen Downloading the authentication credential Unauthorized operation Unlocking a user account Locking a user account Locking the screen Exporting user information Exporting a user group Exporting a role

Audit Log	Operation Type	Operation
	Tenant management	Saving the static configuration Adding a tenant Deleting a tenant Associating a service with a tenant Deleting a service from a tenant Configuring resources Creating resources Deleting resources Adding a resource pool Modifying a resource pool Deleting a resource pool Restoring tenant data

Audit Log	Operation Type	Operation
	Cluster management	Starting a cluster Stopping a cluster Saving configurations Synchronizing cluster configurations Customizing cluster monitoring indicators Saving monitoring thresholds Downloading a client configuration file Configuring the northbound API Configuring the northbound SNMP API Creating a threshold template Deleting a threshold template Applying a threshold template Saving cluster monitoring configuration data Exporting configuration data Importing cluster configuration data Exporting an installation template Modifying a threshold template Canceling the application of a threshold template Masking alarms Sending an alarm Changing the OMS database password Changing the component database password Starting the health check of a cluster

Audit Log	Operation Type	Operation
		Updating the health check configuration Exporting cluster health check results Importing a certificate file Deleting historical health check reports Exporting historical health check reports Customizing report monitoring indicators Exporting report monitoring data Customizing monitoring indicators for static resource pools Exporting monitoring data of a static resource pool
	Service management	Starting a service Stopping a service Synchronizing service configurations Refreshing a service queue Customizing service monitoring indicators Restarting a service Exporting service monitoring data Importing service configuration data Starting the health check of a service Exporting service health check results Configuring the service Uploading a configuration file Downloading a configuration file

Audit Log	Operation Type	Operation
	Instance management	Synchronizing instance configurations Commissioning an instance Decommissioning an instance Starting an instance Stopping an instance Customizing instance monitoring indicators Restarting an instance Exporting instance monitoring data Importing instance configuration data
	Host management	Setting a node rack Starting all roles Stopping all roles Isolating a host Canceling host isolation Customizing host monitoring indicators Exporting host monitoring data Starting the health check of a host Exporting the health check result of a host

Audit Log	Operation Type	Operation
	Maintenance management	Exporting alarms Clearing alarms Exporting events Clearing alarms in batches Clearing alarm through SNMP Adding a trap target through SNMP Deleting a trap target through SNMP Checking alarms through SNMP Synchronizing alarms through SNMP Modifying audit dump configurations Exporting audit logs Collecting log files Downloading log files Uploading a file Deleting an uploaded file Creating a backup task Executing a backup task Stopping a backup task Deleting a backup task Modifying a backup task Locking a backup task Unlocking a backup task Creating a restoration task Executing a backup restoration task Stopping a restoration task Retrying a restoration task Deleting a restoration task

Table 11-21 Component audit information

Audit Log	Operation Type	Operation
DBService audit log	Maintenance management	Performing backup restoration operations
HBase audit log	Data definition language (DDL) statement	Creating a table Deleting a table Modifying a table Adding a column family Modifying a column family Deleting a column family Enabling a table Disabling a table Modify the user information Changing a password User login
	Data manipulation language (DML) statement	Putting data (to the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables) Deleting data (from the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables) Checking and putting data (to the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables) Checking and deleting data (from the **hbase:meta**, **_ctmeta_**, and **hbase:acl** tables)
	Permission control	Assigning permissions to a user Canceling permission assigning

Audit Log	Operation Type	Operation
Hive audit logs	Metadata operation	Defining metadata, such as creating databases and tables Deleting metadata, such as deleting databases and tables Modifying metadata, such as adding columns and renaming tables Importing and exporting metadata
	Data maintenance	Loading data to a table Inserting data into a table
	Permissions management	Creating or deleting roles Granting/Reclaiming roles Granting/Reclaiming permissions
HDFS audit log	Permissions management	Managing permissions on files or folders Managing permissions on owner information files or folders
	File operation	Creating a folder Creating a file Opening a file Appending file content Changing a file name Deleting a file or folder Setting time property of a file Setting the number of file copies Merging files Checking the file system File links

Audit Log	Operation Type	Operation
MapReduce audit log	Application running	Starting a Container request Stopping a Container request After Container request is completed, the status of the request is displayed as succeeded. After Container request is completed, the status of the request is displayed as failed. After Container request is completed, the status of the request is displayed as suspended. Submitting a task Ending a task
LdapServer audit log	Maintenance management	Adding an operating system user Adding a user group Adding a user to user group Deleting a user Deleting a group
KrbServer audit log	Maintenance management	Changing the password of a Kerberos account Adding a Kerberos account Deleting a Kerberos account Authenticating a user
Loader audit log	Security management	User login
	Metadata management	Querying connector information Querying a framework Querying step information

Audit Log	Operation Type	Operation
	Managing data source connections	Querying a data source connection Adding a data source connection Updating a data source connection Deleting a data source connection Activating a data source connection Disabling a data source connection
	Job management	Querying a job Creating a Job Updating a Job Deleting a job Activating a job Disabling a job Querying all execution records of a job Querying the latest execution record of a job Submitting a job Stopping a job
Hue audit log	Service startup	Starting Hue
	User operation	User login User logout
	Task operation	Creating a job Modifying a job Deleting a job Submitting a task Saving a task Updating the status of a task
ZooKeeper audit log	Permissions management	Setting the access permission to Znode
	Znode operation	Creating a Znode Deleting a Znode Configuring Znode data

Audit Log	Operation Type	Operation
Storm audit log	Nimbus	Submitting a topology Stopping a topology Reallocating a topology Deactivating a topology Activating a topology
	UI	Stopping a topology Reallocating a topology Deactivating a topology Activating a topology

MRS audit logs are stored in the database. You can view and export audit logs on the **Audit** page.

The following table lists the directories to store component audit logs. Audit log files of some components are stored in **/var/log/Bigdata/audit**, such as HDFS, HBase, MapReduce, Hive, Hue, Yarn, Storm, and ZooKeeper. The component audit logs are automatically compressed and backed up to **/var/log/Bigdata/audit/bk** at 03: 00 every day. A maximum of latest 90 compressed backup files are retained, and the backup time cannot be changed.

Audit log files of other components are stored in the component log directory.

Table 11-22 Directory for storing component audit logs

Component	Audit Log Directory
DBService	/var/log/Bigdata/audit/dbservice/dbservice_audit.log
HDFS	/var/log/Bigdata/audit/hdfs/nn/hdfs-audit-namenode.log /var/log/Bigdata/audit/hdfs/dn/hdfs-audit-datanode.log /var/log/Bigdata/audit/hdfs/jn/hdfs-audit-journalnode.log /var/log/Bigdata/audit/hdfs/zkfc/hdfs-audit-zkfc.log /var/log/Bigdata/audit/hdfs/httpfs/hdfs-audit-httpfs.log /var/log/Bigdata/audit/hdfs/router/hdfs-audit-router.log
MapReduce	/var/log/Bigdata/audit/mapreduce/jobhistory/mapred-audit-jobhistory.log

Component	Audit Log Directory
Hive	/var/log/Bigdata/audit/hive/hiveserver/hive-audit.log /var/log/Bigdata/audit/hive/metastore/metastore-audit.log /var/log/Bigdata/audit/hive/webhcat/webhcat-audit.log
Loader	/var/log/Bigdata/loader/audit/default.audit
Hue	/var/log/Bigdata/audit/hue/hue-audits.log
ZooKeeper	/var/log/Bigdata/audit/zookeeper/quorumpeer/zk-audit-quorumpeer.log
Spark	/var/log/Bigdata/audit/spark/jdbcserver/jdbcserver-audit.log /var/log/Bigdata/audit/spark/jobhistory/jobhistory-audit.log
Yarn	/var/log/Bigdata/audit/yarn/rm/yarn-audit-resourcemanager.log /var/log/Bigdata/audit/yarn/nm/yarn-audit-nodemanager.log
Storm	/var/log/Bigdata/audit/storm/nimbus/audit.log /var/log/Bigdata/audit/storm/ui/audit.log

11.6.2 Manager Log List

Log Description

Log path: The default storage path of Manager log files is **/var/log/Bigdata/***Manager component*.

- ControllerService: **/var/log/Bigdata/controller/** (operation & maintenance system (OMS) installation and run logs)
- Httpd: **/var/log/Bigdata/httpd** (httpd installation and run logs)
- logman: **/var/log/Bigdata/logman** (log packaging tool logs)
- NodeAgent: **/var/log/Bigdata/nodeagent** (NodeAgent installation and run logs)
- okerberos: **/var/log/Bigdata/okerberos** (okerberos installation and run logs)
- oldapserver: **/var/log/Bigdata/oldapserver** (oldapserver installation and run logs)
- MetricAgent: **/var/log/Bigdata/metric_agent** (MetricAgent run log)
- omm: **/var/log/Bigdata/omm** (omm installation and run logs)
- timestamp: **/var/log/Bigdata/timestamp** (NodeAgent startup time logs)
- tomcat: **/var/log/Bigdata/tomcat** (Web process logs)

- Patch: **/var/log/Bigdata/patch** (patch installation log)
- Sudo: **/var/log/Bigdata/sudo** (sudo script execution log)
- OS: **/var/log/**_message file_ (OS system log)
- OS Performance: **/var/log/osperf** (OS performance statistics log)
- OS Statistics: **/var/log/osinfo/statistics** (OS parameter configuration log)

Log archiving rule:

The automatic compression and archiving function is enabled for Manager logs. By default, when the size of a log file exceeds 10 MB, the log file is automatically compressed. The naming rule of a compressed log file is as follows: <*Original log name*>-<*yyyy-mm-dd_hh-mm-ss*>.[*ID*].**log.zip** A maximum of 20 latest compressed files are reserved.

Table 11-23 Manager logs

Type	Log File Name	Description
Controller run log	controller.log	Log that records component installation, upgrade, patch installation, configuration, monitoring, alarms, and routine O&M operations
	controller_client.log	Run log of the Representational State Transfer (REST) API
	acs.log	ACS run log file
	acs_spnego.log	spnego user log in ACS
	aos.log	AOS run log
	plugin.log	AOS plug-in log
	backupplugin.log	Log that records the backup and restoration operations
	controller_config.log	Configuration run log
	controller_nodesetup.log	Controller loading task log
	controller_root.log	System log of the Controller process
	controller_trace.log	Log that records the remote procedure call (RPC) communication between Controller and NodeAgent

Type	Log File Name	Description
	controller_monitor.log	Monitoring log
	controller_fsm.log	State machine log
	controller_alarm.log	Controller alarm log
	controller_backup.log	Controller backup and recovery log
	install.log, distributeAdapterFiles.log, install_os_optimization.log	OMS installation log
	oms_ctl.log	OMS startup and stop log
	installntp.log	NTP installation log
	modify_manager_param.log	Manager parameter modification log
	backup.log	OMS backup script run log
	supressionAlarm.log	Alarm script run log
	om.log	OM certificate generation log
	backupplugin_ctl.log	Startup log of the backup and restoration plug-in process
	getLogs.log	Run log of the collection log script
	backupAuditLogs.log	Run log of the audit log backup script
	certStatus.log	Log that records regular certificate checks
	distribute.log	Certificate distribution log
	ficertgenetrate.log	Certificate replacement logs, including logs of level-2 certificates, CAS certificates, and httpd certificates
	genPwFile.log	Log that records the generation of certificate password files

Type	Log File Name	Description
	modifyproxyconf.log	Log that records the modification of the HTTPD proxy configuration
	importTar.log	Log that records the process of importing certificates into the trust library
Httpd	install.log	Httpd installation log
	access_log, error_log	Httpd run log
logman	logman.log	Log packaging tool log
NodeAgent	install.log, install_os_optimization.log	NodeAgent installation log
	installntp.log	NTP installation log
	start_ntp.log	NTP startup log
	ntpChecker.log	NTP check log
	ntpMonitor.log	NTP monitoring log
	heartbeat_trace.log	Log that records heartbeats between NodeAgent and Controller
	alarm.log	Alarm log
	monitor.log	Monitoring log
	nodeagent_ctl.log, start-agent.log	NodeAgent startup log
	agent.log	NodeAgent run log
	cert.log	Certificate log
	agentplugin.log	Agent plug-in running status monitoring log
	omaplugin.log	OMA plug-in run log
	diskhealth.log	Disk health check log
	supressionAlarm.log	Alarm script run log
	updateHostFile.log	Host list update log
	collectLog.log	Run log of the node log collection script

Type	Log File Name	Description
	host_metric_collect.log	Host index collection run log
	checkfileconfig.log	Run log file of file permission check
	entropycheck.log	Entropy check run log
	timer.log	Log of periodic node scheduling
	pluginmonitor.log	Component monitoring plug-in log
	agent_alarm_py.log	Log that records alarms upon insufficient NodeAgent file permission
okerberos	addRealm.log, modifyKerberosRealm.log	Domain handover log
	checkservice_detail.log	Okerberos health check log
	genKeytab.log	keytab generation log
	KerberosAdmin_genConfigDetail.log	Run log that records the generation of kadmin.conf when starting the kadmin process
	KerberosServer_genConfigDetail.log	Run log that records the generation of krb5kdc.conf when starting the krb5kdc process
	oms-kadmind.log	Run log of the kadmin process
	oms_kerberos_install.log, postinstall_detail.log	Okerberos installation log
	oms-krb5kdc.log	Run log of the krbkdc process
	start_detail.log	Okerberos startup log
	realmDataConfigPro-cess.log	Log rollback for domain handover failure
	stop_detail.log	Okerberos stop log
oldapserver	ldapserver_backup.log	Oldapserver backup log

Type	Log File Name	Description
	ldapserver_chk_service.log	Oldapserver health check log
	ldapserver_install.log	Oldapserver installation log
	ldapserver_start.log	Oldapserver startup log
	ldapserver_status.log	Log that records the status of the Oldapserver process
	ldapserver_stop.log	Oldapserver stop log
	ldapserver_wrap.log	Oldapserver service management log
	ldapserver_uninstall.log	Oldapserver uninstallation log
	restart_service.log	Oldapserver restart log
	ldapserver_unlockUser.log	Log that records information about unlocking LDAP users and managing accounts
omm	omsconfig.log	OMS configuration log
	check_oms_heartbeat.log	OMS heartbeat log
	monitor.log	OMS monitoring log
	ha_monitor.log	HA_Monitor operation log
	ha.log	HA operation log
	fms.log	Alarm log
	fms_ha.log	HA alarm monitoring log
	fms_script.log	Alarm control log
	config.log	Alarm configuration log
	iam.log	IAM log
	iam_script.log	IAM control log
	iam_ha.log	IAM HA monitoring log
	config.log	IAM configuration log
	operatelog.log	IAM operation log

Type	Log File Name	Description
	heartbeatcheck_ha.log	OMS heartbeat HA monitoring log
	install_oms.log	OMS installation log
	pms_ha.log	HA monitoring log
	pms_script.log	Monitoring control log
	config.log	Monitoring configuration log
	plugin.log	Monitoring plug-in run log
	pms.log	Monitoring log
	ha.log	HA run log
	cep_ha.log	CEP HA monitoring log
	cep_script.log	CEP control log
	cep.log	CEP log
	config.log	CEP configuration log
	omm_gaussdba.log	GaussDB HA monitoring log
	gaussdb-<SERIAL>.log	GaussDB run log
	gs_ctl-<DATE>.log	GaussDB control log archive log
	gs_ctl-current.log	GaussDB control log
	gs_guc-current.log	GaussDB operation log
	encrypt.log	Omm encryption log
	omm_agent_ctl.log	OMA control log
	oma_monitor.log	OMA monitoring log
	install_oma.log	OMA installation log
	config_oma.log	OMA configuration log
	omm_agent.log	OMA run log
	acs.log	ACS resource log
	aos.log	AOS resource log
	controller.log	Controller resource log

Type	Log File Name	Description
	feed_watchdog.log	feed_watchdog resource log
	floatip.log	Floating IP address resource log
	ha_ntp.log	NTP resource log
	httpd.log	Httpd resource log
	okerberos.log	Okerberos resource log
	oldap.log	OLdap resource log
	tomcat.log	Tomcat resource log
	send_alarm.log	Run log of the HA alarm sending script of the management node
timestamp	restart_stamp	NodeAgent start time log
tomcat	cas.log, localhost_access_cas_log.log	CAS run log
	catalina.log, catalina.out, host-manager.log, localhost.log, manager.log	Tomcat run log
	localhost_access_web_log.log	Log that records the access to REST APIs of FusionInsight Manager
	web.log	Run log of the web process
	northbound_ftp_sftp.log, snmp.log	Northbound log
watchdog	watchdog.log, feed_watchdog.log	watchdog run log
patch	oms_installPatch.log	OMS patch installation log
	agent_installPatch.log	Agent patch installation log
	agent_uninstallPatch.log	Agent patch uninstallation log
	NODE_AGENT_restoreFile.log	Agent patch restoration log

Type	Log File Name	Description
	NODE_AGENT_updateFile.log	Agent patch update log
	OMA_restoreFile.log	OMA patch restoration file log
	OMA_updateFile.log	OMA patch update file log
	CONTROLLER_restoreFile.log	CONTROLLER patch restoration file log
	CONTROLLER_updateFile.log	CONTROLLER patch update file log
	OMS_restoreFile.log	OMS patch restoration file log
	oms_uninstallPatch.log	OMS patch uninstallation log
	OMS_updateFile.log	OMS patch update file log
	createStackConf.log, decompress.log, decompress_OMS.log, distrExtractPatchO-nOMS.log, slimReduction.log, switch_adapter.log	Patch installation log
sudo	sudo.log	Sudo script execution log

Log Levels

Table 11-24 describes the log levels provided by Manager. The priorities of log levels are FATAL, ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 11-24 Log levels

Level	Description
FATAL	Logs of this level record fatal error information about the current event processing that may result in a system crash.
ERROR	Logs of this level record error information about the current event processing, which indicates that system running is abnormal.

Level	Description
WARN	Abnormal information about the current event processing. These abnormalities will not result in system faults.
INFO	Normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

Log Formats

The following table lists the Manager log formats.

Table 11-25 Log formats

Type	Component	Format	Example
Controller, Httpd, logman, NodeAgent, okerberos, oldapserver, omm, tomcat, upgrade	Controller, Httpd, logman, NodeAgent, okerberos, oldapserver, omm, tomcat, upgrade	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2015-06-30 00:37:09,067 INFO [pool-1-thread-1] Completed Discovering Node. com.xxx.hadoop.om.controller.tasks.nodesetup.DiscoverNodeTask.execute(DiscoverNodeTask.java:299)

11.6.3 Viewing and Exporting Audit Logs

Scenario

This section describes how to view and export audit logs on MRS Manager. The audit logs can be used to trace security events, locate fault causes, and determine responsibilities.

The system record the following log information:

- User activity information, such as user login and logout, system user information modification, and system user group information modification
- User operation instruction information, such as cluster startup, stop, and software upgrade.

Procedure

- Viewing audit logs

a. On MRS Manager, click **Audit** to view the default audit logs.

   If the audit content of an audit log contains more than 256 characters, click the expand button of the audit log to expand the audit details. Click **Log File** to download the complete file and view the information.

   - By default, records are sorted in descending order by the **Occurred** column. You can click **Operation Type**, **Severity**, **Occurred**, **User**, **Host**, **Service**, **Instance**, or **Operation Result** to change the sorting mode.
   - All alarms of the same severity can be filtered by **Severity**. The results include cleared and uncleared alarms.

   Exported audit logs contain the following information:

   - **Sno**: indicates the number of audit logs generated by MRS Manager. The number is incremented by 1 when a new audit log is generated.
   - **Operation Type**: indicates the operation type of a user operation. There are nine scenarios: **Alarm**, **Auditlog**, **Backup And Restoration**, **Cluster**, **Collect Log**, **Host**, **Service**, **Tenant** and **User_Manager**. **User_Manager** is supported only in clusters with Kerberos authentication enabled. Each scenario contains different operation types. For example, **Alarm** includes **Export alarms**; **Cluster** includes **Start cluster**, and **Tenant** include **Add tenant**.
   - **Severity**: indicates the security level of each audit log, including **Critical**, **Major**, **Minor** and **Informational**.
   - **Start Time**: indicates the time when the operation starts. The time is UTC/GMT+08:00.
   - **End Time**: indicates the time when the operation ends. The time is UTC/GMT+08:00.
   - **User IP Address**: indicates the IP address used by a user to perform operations.
   - **User**: indicates the name of the user who performs the operation.
   - **Host**: indicates the node where the user operation is performed. The information is not saved if the operation does not involve a node.
   - **Service**: indicates the service in the cluster where the user operation is performed. The information is not saved if the operation does not involve a service.
   - **Instance**: indicates the role instance in the cluster where the user operation is performed. The information is not saved if the operation does not involve a role instance.
   - **Operation Result**: indicates the operation result, including **Successful**, **Failed** and **Unknown**.
   - **Content**: indicates execution information of the user operation.

b. Click **Advanced Search**. In the search area, set search criteria and click **Search** to view audit logs of the specified type. Click **Reset** to clear the search criteria.

NOTE

Start Time and **End Time** specify the start time and end time of the time range. You can search for alarms generated within the time range.

- Exporting audit logs
  a. In the audit log list, click **Export All** to export all logs.
  b. In the audit log list, select the check box of a log and click **Export** to export the log.

11.6.4 Exporting Service Logs

Scenario

This section describes how to export logs generated by each service role from MRS Manager.

Prerequisites

- You have obtained the access key ID (AK) and secret access key (SK) of the account.
- A parallel file system has been created in OBS.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 Click **Export Log** under **Maintenance**.

Step 3 Set a service for **Service**. Set **Host** to the IP address of the host where the service is deployed. Select the corresponding time for **Start Time** and **End Time**.

Step 4 In **Export To**, select a path for saving logs. This parameter is available only for clusters with Kerberos authentication enabled.

- **Local PC**: indicates that logs are saved to the local environment. Then go to **Step 8**.
- **OBS**: indicates that logs are saved to OBS. This is the default option. Then go to **Step 5**.

Step 5 Set **OBS Path** to the path for storing service logs on OBS.

The value must be a complete path and cannot start with a slash (/). The path can be nonexistent and will be automatically created by the system. The full path of OBS can contain a maximum of 900 bytes.

Step 6 In **Bucket**, enter the name of the created OBS file system.

Step 7 Set **AK** and **SK** to the access key ID and secret access key of the user.

Step 8 Click **OK**.

----End

11.6.5 Configuring Audit Log Exporting Parameters

Scenario

If MRS audit logs are stored in the system for a long time, the disk space of the data directory may be insufficient. Therefore, you can set export parameters to automatically export audit logs to a specified directory on the OBS server timely, facilitating audit log management.

NOTE

Audit logs exported to the OBS server include service audit logs and management audit logs.

- Service audit logs are automatically compressed and stored in the **/var/log/Bigdata/audit/bk/** directory on the active management node at 03:00 every day. The file name format is <*yyyy-MM-dd_HH-mm-ss*>**.tar.gz**. By default, a maximum of seven log files can be stored. If more than seven log files are stored, the system automatically deletes the log files generated seven days ago.
- The data range of management audit logs exported to OBS each time is from the last date when the logs are successfully exported to OBS to the date when the task is executed. When the number of management audit logs reaches 100,000, the system automatically dumps the first 90,000 audit logs to a local file and retains 10,000 audit logs in the database. The dumped log files are saved in the **${BIGDATA_DATA_HOME}/dbdata_om/dumpData/iam/operatelog** directory on the active management node. The file name format is **OperateLog_store_***YY_MM_DD_HH_MM_SS***.csv**. A maximum of 50 historical audit log files can be saved.

Prerequisites

- You have obtained the access key ID (AK) and secret access key (SK) of the account.
- A parallel file system has been created in OBS.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 Choose **Export Audit Log** under **Maintenance**.

Table 11-26 Parameters for exporting audit logs

Parameter	Value	Description
Start Time	7/24/2017 09:00:00 (example value)	(Mandatory) Specifies the start time for exporting audit logs.
Period (days)	1 day (example value)	(Mandatory) Specifies the interval for exporting audit logs. The interval ranges from 1 to 5 days.
Bucket	mrs-bucket (example value)	(Mandatory) Specifies the name of the OBS file system to which audit logs are exported.

Parameter	Value	Description
OBS path	**/opt/omm/oms/auditLog** (example value)	(Mandatory) Specifies the OBS path to which audit logs are exported.
AK	*XXX* (example value)	(Mandatory) Specifies the user's access key ID.
SK	*XXX* (example value)	(Mandatory) Specifies the user's secret access key.

NOTE

Audit logs are stored in **service_auditlog** and **manager_auditlog** on OBS, which are used to store service audit logs and management audit logs, respectively.

----End

11.7 Health Check Management

11.7.1 Performing a Health Check

Scenario

To ensure that cluster parameters, configurations, and monitoring are correct and that the cluster can run stably for a long time, you can perform a health check during routine maintenance.

NOTE

A system health check includes MRS Manager, service-level, and host-level health checks:

- MRS Manager health checks focus on whether the unified management platform can provide management functions.
- Service-level health checks focus on whether components can provide services properly.
- Host-level health checks focus on whether host indicators are normal.

The system health check includes three types of check items: health status, related alarms, and customized monitoring indicators for each check object. The health check results are not always the same as the **Health Status** on the portal.

Procedure

- Manually perform the health check for all services.
  a. Click **Services** and select the target service.
  b. Choose **More** > **Start Service Health Check** to start the health check for the service.

NOTE

- The cluster health check includes Manager, service, and host status checks.
- To perform cluster health checks, you can also choose **System** > **Check Health Check** > **Start Cluster Health Check** on MRS Manager.
- To export the health check result, click **Export Report** in the upper left corner.

- Manually perform the health check for a service.
  a. Click **Services**. In the services list, click the desired service name.
  b. Choose **More** > **Start Service Health Check** to start the health check for the service.
- Manually perform the health check for a host.
  a. Click **Hosts**.
  b. Select the check box of the host for which you want to check the health status.
  c. Choose **More** > **Start Host Health Check** to start the health check for the host.
- Automatically performing a health check
  a. Click **System**.
  b. Click **Check Health Status** under **Maintenance**.
  c. Click **Configure Health Check** to configure automatic health check items.

     Periodic Health Check: specifies whether to enable automatic health check. The **Periodic Health Check** function is disabled by default. You can click to enable the function and select **Daily**, **Weekly**, or **Monthly** based on management requirements.
  d. Click **OK** to save the settings. The **Health check configuration saved successfully** is displayed in the upper right corner.

11.7.2 Viewing and Exporting a Health Check Report

Scenario

You can view the health check result in MRS Manager and export the health check results for further analysis.

NOTE

A system health check includes MRS Manager, service-level, and host-level health checks:

- MRS Manager health checks focus on whether the unified management platform can provide management functions.
- Service-level health checks focus on whether components can provide services properly.
- Host-level health checks focus on whether host indicators are normal.

The system health check includes three types of check items: health status, related alarms, and customized monitoring indicators for each check object. The health check results are not always the same as the **Health Status** on the portal.

Prerequisites

You have performed a health check.

Procedure

Step 1 Click **Services**.

Step 2 Choose **More > View Cluster Health Check Report** to view the health check report of a cluster.

Step 3 Click **Export Report** on the health check report pane to export the report and view detailed information about check items.

NOTE

For details about how to rectify the faults of the check items, see **DBService Health Check Indicators** to **ZooKeeper Health Check Indicators**.

----End

11.7.3 Configuring the Number of Health Check Reports to Be Reserved

Scenario

Health check reports of MRS clusters, services, and hosts may vary with the time and scenario. You can modify the number of health check reports to be reserved on MRS Manager for later comparison.

This setting is valid for health check reports of clusters, services, and hosts. Report files are saved in **$BIGDATA_DATA_HOME/Manager/healthcheck** on the active management node by default and are automatically synchronized to the standby management node.

Prerequisites

Users have specified service requirements and planned the save time and health check frequency, and the disk space of the active and standby management nodes is sufficient.

Procedure

Step 1 Choose **System** > **Check Health Status** > **Configure Health Check**.

Step 2 Set **Max. Number of Health Check Reports** to the number of health check reports to be reserved. The value ranges from 1 to 100. The default value is 50.

Step 3 Click **OK** to save the settings. The **Health check configuration saved successfully** is displayed in the upper right corner.

----End

11.7.4 Managing Health Check Reports

Scenario

On MRS Manager, users can manage historical health check reports, for example, viewing, downloading, and deleting historical health check reports.

Procedure

- Download a specified health check report.
  a. Choose **System** > **Check Health Status**.
  b. Locate the row that contains the target health check report and click **Download** to download the report file.
- Download specified health check reports in batches.
  a. Choose **System** > **Check Health Status**.
  b. Select multiple health check reports and click **Download File** to download them.
- Delete a specified health check report.
  a. Choose **System** > **Check Health Status**.
  b. Locate the row that contains the target health check report and click **Delete** to delete the report file.
- Delete specified health check reports in batches.
  a. Choose **System** > **Check Health Status**.
  b. Select multiple health check reports and click **Delete File** to delete them.

11.7.5 DBService Health Check Indicators

Service Health Check

Indicator: Service Status

Description: This indicator is used to check whether the DBService service status is normal. If the status is abnormal, the service is unhealthy.

Handling method: If the indicator is abnormal, rectify the fault by referring to ALM-27001.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist on the host. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.6 Flume Health Check Indicators

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Flume service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If the indicator is abnormal, rectify the fault by referring to ALM-24000.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist on the host. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.7 HBase Health Check Indicators

Normal RegionServer Count

Indicator: Normal RegionServer Count

Description: This indicator is used to check the number of RegionServers that are running properly in an HBase cluster.

Recovery Guide: If the indicator is abnormal, check whether the status of RegionServer is normal. If the status is abnormal, resolve the problem and check that the network is normal.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the HBase service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If the indicator is abnormal, check whether the status of HMaster and RegionServer is normal. If the status is abnormal, resolve the problem. Then, check whether the status of the ZooKeeper service is faulty. On the HBase client, check whether the data in the HBase table can be correctly read and locate the data reading failure cause. Handle the alarm following instructions in the alarm processing document.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.8 Host Health Check Indicators

Swap Usage

Indicator: Swap Usage

Description: Swap usage of the system. The value is calculated using the following formula: Swap usage = Used swap size/Total swap size. Assume that the

current threshold is set to 75.0%. If the usage of the file handles in the system exceeds the threshold, the system is unhealthy.

Recovery Guide:

1. Check the swap usage of the node.

   Log in to the unhealthy node and run the **free -m** command to check the total swap space and used swap space. If the swap space usage exceeds the threshold, go to **2**.
2. If the swap usage exceeds the threshold, you are advised to expand the system capacity, for example, add nodes.

Host File Handle Usage

Indicator: Host File Handle Usage

Description: This indicator indicates the file handle usage in the system. Host file handle usage = Number of used handles/Total number of handles. If the usage exceeds the threshold, the system is unhealthy.

Recovery Guide:

1. Check the file handle usage of the host.

   Log in to the unhealthy node and run the **cat /proc/sys/fs/file-nr** command. In the command output, the first and third columns indicate the number of used handles and the total number of handles, respectively. If the usage exceeds the threshold, go to **2**.
2. If the file handle usage of the host exceeds the threshold, you are advised to check the system and analyze the file handle usage.

NTP Offset

Indicator: NTP Offset

Description: This indicator indicates the NTP time offset. If the time deviation exceeds the threshold, the system is unhealthy.

Recovery Guide:

1. Check the NTP time offset.

   Log in to the unhealthy node and run the **/usr/sbin/ntpq -np** command to view the information. In the command output, the **Offset** column indicates the time offset. If the time offset is greater than the threshold, go to **2**.
2. If the indicator is abnormal, check whether the clock source configuration is correct. Contact O&M personnel.

Average Load

Indicator: Average Load

Description: Average system load, indicating the average number of processes in the running queue in a specified period. The system average load is calculated using the load value obtained by the uptime command. Calculation method: (Load of 1 minute + Load of 5 minutes + Load of 15 minutes)/(3 x Number of CPUs).

Assume that the current threshold is set to 2. If the average load exceeds 2, the system is unhealthy.

Recovery Guide:

1. Log in to the unhealthy node and run the **uptime** command. The last three columns in the command output indicate the load in 1 minute, 5 minutes, and 15 minutes, respectively. If the average system load exceeds the threshold, go to **2**.
2. If the system average load exceeds the threshold, you are advised to perform system capacity expansion, such as adding nodes.

D State Process

Indicator: D State Process

Description: This indicator indicates the unstoppable sleep process, that is, the process in the D state. A process that is in the D state is waiting for I/O, such as disk I/O and network I/O, and experiences an I/O exception. If any process in the D state exists in the system, the system is unhealthy.

Recovery Guide: If the indicator is abnormal, the system generates an alarm. You are advised to handle the alarm by referring to ALM-12028.

Hardware Status

Indicator: Hardware Status

Description: This indicator is used to check the system hardware status, including the CPU, memory, disk, power supply, and fan. This indicator obtains related hardware information using **ipmitool sdr elist**. If the hardware status is abnormal, the hardware is unhealthy.

Recovery Guide:

1. Log in to the node where the check result is unhealthy. Run the **ipmitool sdr elist** command to check system hardware status. The last column in the command output indicates the hardware status. If the status is included in the following fault description table, the check result is unhealthy.

Module	Symptom
Processor	IERR Thermal Trip FRB1/BIST failure FRB2/Hang in POST failure FRB3/Processor startup/init failure Configuration Error SM BIOS Uncorrectable CPU-complex Error Disabled Throttled Uncorrectable machine check exception
Power Supply	Failure detected Predictive failure Power Supply AC lost AC lost or out-of-range AC out-of-range, but present Config Error: Vendor Mismatch Config Error: Revision Mismatch Config Error: Processor Missing Config Error: Power Supply Rating Mismatch Config Error: Voltage Rating Mismatch Config Error
Power Unit	240VA power down Interlock power down AC lost Soft-power control failure Failure detected Predictive failure
Memory	Uncorrectable ECC Parity Memory Scrub Failed Memory Device Disabled Correctable ECC logging limit reached Configuration Error Throttled Critical Overtemperature

Module	Symptom
Drive Slot	Drive Fault Predictive Failure Parity Check In Progress In Critical Array In Failed Array Rebuild In Progress Rebuild Aborted
Battery	Low Failed

2. If the indicator is abnormal, contact O&M personnel.

Host Name

Indicator: Host Name

Description: This indicator is used to check whether the host name is set. If the host name is not set, the system is unhealthy. If the indicator is abnormal, you are advised to set the host name properly.

Recovery Guide:

1. Log in to the node where the check result is unhealthy.
2. Run the hostname host name command to change the host name to ensure that the host name is consistent with the planned host name.

   hostname*host name* For example, to change the host name to **Bigdata-OM-01**, run the **hostname Bigdata-OM-01** command.
3. Modify the host name configuration file.

   Run the **vi /etc/HOSTNAME** command to edit the file. Change the file content to **Bigdata-OM-01**. Save the file, and exit.

Umask

Indicator: Umask

Description: This indicator is used to check whether the umask setting of user **omm** is correct. If Umask is not 0077, the system is unhealthy.

Recovery Guide:

1. If the indicator is abnormal, you are advised to set umask of user **omm** to 0077. Log in to the unhealthy node and run the **su - omm** command to switch to user **omm**.
2. Run the **vi ${BIGDATA_HOME}/.om_profile** command and change the value of **umask** to **0077**. Save and exit.

OMS HA Status

Indicator: OMS HA Status

Description: This indicator is used to check whether the OMS two-node cluster resources are normal. You can run the **${CONTROLLER_HOME}/sbin/status-oms.sh** command to view the detailed information about the status of the OMS two-node cluster resources. If any module is abnormal, the OMS is unhealthy.

Recovery Guide:

1. Log in to the active management node and run the **su - omm** command to switch to user **omm**. Run the **${CONTROLLER_HOME}/sbin/status-oms.sh** command to check the OMS status.
2. If floatip, okerberos, and oldap are abnormal, handle the problems by referring to ALM-12002, ALM-12004, and ALM-12005 respectively.
3. If other resources are abnormal, you are advised to view the logs of the faulty modules.

   If controller resources are abnormal, view **/var/log/Bigdata/controller/controller.log** of the faulty node.

   If CEP resources are abnormal, view **/var/log/Bigdata/omm/oms/cep/cep.log** of the faulty node.

   If AOS resources are abnormal, view **/var/log/Bigdata/controller/aos/aos.log** of the faulty node.

   If feed_watchdog resources are abnormal, view **/var/log/Bigdata/watchdog/watchdog.log** of the abnormal node.

   If HTTPD resources are abnormal, view **/var/log/Bigdata/httpd/error_log** of the abnormal node.

   If FMS resources are abnormal, view **/var/log/Bigdata/omm/oms/fms/fms.log** of the abnormal node.

   If PMS resources are abnormal, view **/var/log/Bigdata/omm/oms/pms/pms.log** of the abnormal node.

   If IAM resources are abnormal, view **/var/log/Bigdata/omm/oms/iam/iam.log** of the abnormal node.

   If the GaussDB resource is abnormal, check the **/var/log/Bigdata/omm/oms/db/omm_gaussdba.log** of the abnormal node.

   If NTP resources are abnormal, view **/var/log/Bigdata/omm/oms/ha/scriptlog/ha_ntp.log** of the abnormal node.

   If Tomcat resources are abnormal, view **/var/log/Bigdata/tomcat/catalina.log** of the abnormal node.
4. If the fault cannot be rectified based on the logs, contact O&M personnel and send the collected fault logs.

Checking the Installation Directory and Data Directory

Indicator: Installation Directory and Data Directory Check

Description: This indicator checks the **lost+found** directory in the root directory of the disk partition where the installation directory (**/opt/Bigdata** by default) is located. If the directory contains the files of user **omm**, there are exceptions.

When a node is abnormal, related files are stored in the **lost+found** directory. This indicator is used to check whether files are lost in such scenarios. Check the installation directory (for example, **/opt/Bigdata**) and data directory (for example, **/srv/BigData**). If any files of non-omm users exist in the two directories, the system is unhealthy.

Recovery Guide:

1. Log in to the unhealthy node and run the **su - omm** command to switch to user **omm**. Check whether files or folders of user omm exist in the **lost+found** directory.

   If the **omm** user file exists, you are advised to restore it and check again. If the **omm** user file does not exist, go to **2**.
2. Check the installation directory and data directory. Check whether the files or folders of other users exist in the installation directory and data directory. If the files and folders are manually generated temporary files, you are advised to delete them and check again.

CPU Usage

Indicator: CPU Usage

Description: This indicator is used to check whether the CPU usage exceeds the threshold. If the disk usage exceeds the threshold, the system is unhealthy.

Recovery Guide: If the indicator is abnormal, the system generates an alarm. You are advised to handle the alarm by referring to ALM-12016.

Memory Usage

Indicator: Memory Usage

Description: This indicator is used to check whether the memory usage exceeds the threshold. If the disk usage exceeds the threshold, the system is unhealthy.

Recovery Guide: If the indicator is abnormal, the system generates an alarm. You are advised to handle the alarm by referring to ALM-12018.

Host Disk Usage

Indicator: Host Disk Usage

Description: This indicator is used to check whether the host disk usage exceeds the threshold. If the disk usage exceeds the threshold, the system is unhealthy.

Recovery Guide: If the indicator is abnormal, the system generates an alarm. You are advised to handle the alarm by referring to ALM-12017.

Host Disk Write Rate

Indicator: Host Disk Write Rate

Description: This indicator is used to check the disk write rate of a host. The write rate of the host disk may vary according to the service scenario. Therefore, the value of this indicator reflects only the specified value. You need to determine whether the indicator is normal in specified service scenarios.

Recovery Guide: Determine whether the current disk write rate is normal based on the service scenario.

Host Disk Read Rate

Indicator: Host Disk Read Rate

Description: This indicator is used to check the disk read rate of a host. The read rate of the host disk may vary by service scenario. Therefore, the value of this indicator reflects only the specified value. You need to determine whether the indicator is normal in specified service scenarios.

Recovery Guide: Determine whether the current disk read rate is normal based on the service scenario.

Host Service Plane Network Status

Indicator: Host Service Plane Network Status

Description: This indicator is used to check the connectivity of the service plane network of the cluster host. If the hosts are disconnected, the cluster is unhealthy.

Recovery Guide: If the single-plane networking is used, check the IP address of the single plane. For a dual-plane network, the operation procedure is as follows:

1. Check the network connectivity between the service plane IP addresses of the active and standby management nodes.

   If the network is abnormal, go to **3**.

   If the network is normal, go to **2**.
2. Check the network connectivity between the IP address of the active management node and the IP address of the abnormal node in the cluster.
3. If the network is disconnected, contact O&M personnel to rectify the network fault to ensure that the network meets service requirements.

Host Status

Indicator: Host Status

Description: This indicator is used to check whether the host status is normal. If a node is faulty, the host is unhealthy.

Recovery Guide: If the indicator is abnormal, rectify the fault by referring to ALM-12006.

Alarm Check

Indicator: Alarm Check

Description: This indicator is used to check whether alarms exist on the host. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.9 HDFS Health Check Indicators

Average Packet Sending Time

Indicator: Average Packet Sending Time

Description: This indicator is used to collect statistics on the average time for the DataNode in the HDFS to execute SendPacket each time. If the average time is greater than 2,000,000 ns, the DataNode is unhealthy.

Recovery Guide: If the indicator is abnormal, check whether the network speed of the cluster is normal and whether the memory or CPU usage is too high. Check whether the HDFS load in the cluster is high.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the HDFS service status is normal. If a node is faulty, the host is unhealthy.

Recovery Guide: If the indicator is abnormal, check whether the health status of the KrbServer, LdapServer and ZooKeeper services are faulty. If yes, rectify the fault. Then, check whether the file writing failure is caused by HDFS SafeMode ON. Use the client to check whether data cannot be written into HDFS and locate the cause of the HDFS data writing failure. Handle the alarm following instructions in the alarm processing document.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.10 Hive Health Check Indicators

Maximum Number of Sessions Allowed by HiveServer

Indicator: Maximum Number of Sessions Allowed by HiveServer

Description: This indicator is used to check the maximum number of sessions that can be connected to Hive.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Number of Sessions Connected to HiveServer

Indicator: Number of Sessions Connected to HiveServer

Description: This indicator is used to check the number of Hive connections.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Hive service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist on the host. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.11 Kafka Health Check Indicators

Number of Available Broker Nodes

Indicator: Number of Brokers

Description: This indicator is used to check the number of available Broker nodes in a cluster. If the number of available Broker nodes in a cluster is less than 2, the cluster is unhealthy.

Recovery Guide: If the indicator is abnormal, go to the Kafka service instance page and click the host name of the unavailable Broker instance. View the host health status in the **Overview** area. If the host health status is **Good**, rectify the fault by referring to the alarm handling suggestions in **Process Fault**. If the status is not **Good**, rectify the fault by referring to the handling procedure of the **Node Fault** alarm.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Kafka service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If the indicator is abnormal, rectify the fault by referring to the alarm "Kafka Service Unavailable".

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.12 KrbServer Health Check Indicators

KerberosAdmin Service Availability

Indicator: KerberosAdmin Service Availability

Description: The system checks the KerberosAdmin service status. If the check result is abnormal, the KerberosAdmin service is unavailable.

Recovery Guide: If the indicator check result is abnormal, the possible cause is that the node where the KerberosAdmin service is located is faulty or the SlapdServer service is unavailable. During the KerberosAdmin service recovery, try the following operations:

1. Check whether the node where the KerberosAdmin service locates is faulty.
2. Check whether the SlapdServer service is unavailable.

KerberosServer Service Availability

Indicator: KerberosServer Service Availability

Description: The system checks the KerberosServer service status. If the check result is abnormal, the KerberosServer service is unavailable.

Recovery Guide: If the indicator check result is abnormal, the possible cause is that the node where the KerberosServer service is located is faulty or the SlapdServer service is unavailable. During the KerberosServer service recovery, try the following operations:

1. Check whether the node where the KerberosServer service locates is faulty.
2. Check whether the SlapdServer service is unavailable.

Service Health Status

Indicator: Service Status

Description: The system checks the KrbServer service status. If the check result is abnormal, the KrbServer service is unavailable.

Recovery Guide: If the indicator check result is abnormal, the possible cause is that the node where the KrbServer service resides is faulty or the LdapServer service is unavailable. For details, see the handling procedure of ALM-25500.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check the alarm information about the KrbServer service. If any alarms exist, the KrbServer service may be abnormal.

Recovery Guide: If this indicator check result is abnormal, see the related alarm document to handle the alarms.

11.7.13 LdapServer Health Check Indicators

SlapdServer Service Availability

Indicator: SlapdServer Service Availability

Description: The system checks the SlapdServer service status. If the status is abnormal, the SlapdServer service is unavailable.

Recovery Guide: If the indicator check result is abnormal, the possible cause is that the node where the SlapdServer service is located is faulty or the SlapdServer process is faulty. During the SlapdServer service recovery, try the following operations:

1. Check whether the node where the SlapdServer service locates is faulty. For details, see ALM-12006.
2. Check whether the SlapdServer process is normal. For details, see ALM-12007.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check the alarm information about the LdapServer service. If the status is abnormal, the LdapServer service is unavailable.

Recovery Guide: If the indicator check result is abnormal, the possible cause is that the node where the active LdapServer service resides is faulty or the active LdapServer process is faulty. For details, see ALM-25000.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check the alarm information about the LdapServer service. If any alarms exist, the LdapServer service may be abnormal.

Recovery Guide: If this indicator check result is abnormal, see the related alarm document to handle the alarms.

11.7.14 Loader Health Check Indicators

ZooKeeper Health Status

Indicator: ZooKeeper health status

Description: This indicator is used to check whether the ZooKeeper health status is normal. If the status is abnormal, the ZooKeeper service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

HDFS Health Status

Indicator: HDFS health status

Description: This indicator is used to check whether the HDFS health status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

DBService Health Status

Indicator: DBService Health Status

Description: This indicator is used to check whether the DBService health status is normal. If the status is abnormal, the DBService service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Yarn Health Status

Indicator: Yarn health status

Description: This indicator is used to check whether the Yarn health status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

MapReduce Health Status

Indicator: MapReduce Health Status

Description: This indicator is used to check whether the MapReduce health status is normal. If the status is abnormal, the MapReduce service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Loader Process Status

Indicator: Loader Process Status

Description: This indicator is used to check whether the Loader process is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Loader service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist for loader. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.15 MapReduce Health Check Indicators

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the MapReduce service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.16 OMS Health Check Indicators

OMS Status Check

Indicator: OMS Status Check

Description: The OMS status check includes the HA status check and resource status check. The HA status includes **active**, **standby**, and **NULL**, indicating the active node, standby node, and unknown, respectively. The resource status includes normal, abnormal, and NULL. If the HA status is NULL, the HA status is unhealthy. If the resource status is NULL or abnormal, the resource status is unhealthy.

Table 11-27 OMS status description

Name	Description
HA state	**active**: indicates the active node. **standby**: indicates the standby node. **NULL**: unknown

Name	Description
Resource status	**normal**: All resources are normal. **abnormal**: indicates that abnormal resources exist. **NULL**: unknown

Recovery Guide:

1. Log in to the active management node and run the **su - omm** command to switch to user **omm**. Run the **${CONTROLLER_HOME}/sbin/status-oms.sh** command to check the status of OMS.
2. If the HA status is NULL, the system may be restarting. NULL is an intermediate state, and the HA status will automatically change to a normal state.
3. If the resource status is abnormal, certain component resources of FusionInsight Manager are abnormal. Check whether the status of components such as acs, aos cep, controller, feed_watchdog, fms, gaussDB, httpd, iam, ntp, okerberos, oldap, pms, and tomcat component is normal.
4. If any Manager component resource is abnormal, see Manager component status check to rectify the fault.

Manager Component Status Check

Indicator: Manager Component Status Check

Description: This indicator is used to check the running status and HA status of Manager components. The resource running status includes **Normal** and **Abnormal**, and the resource HA status includes **Normal** and **Exception**. Manager components include Acs, Aos, Cep, Controller, feed_watchdog, Floatip, Fms, GaussDB, HeartBeatCheck, httpd, IAM, NTP, Okerberos, OLDAP, PMS, and Tomcat. If the running status and HA status is not Normal, the check result is unhealthy.

Table 11-28 Manager status description

Name	Description
Resource running status:	**Normal**: The system is running properly. **Abnormal**: The running is abnormal. **Stopped**: The task is stopped. **Unknown**: The status is unknown. **Starting**: The process is being started. **Stopping**: The task is being stopped. **Active_normal**: The active node is running properly. **Standby_normal**: The standby node is running properly. **Raising_active**: The node is being promoted to be the active node. **Lowing_standby**: The node is being set to be the standby node. **No_action**: the action does not exist. **Repairing**: The disk is being repaired. **NULL**: unknown
Resource HA status	**Normal**: the status is normal. **Exception**: indicates a fault. **Non_steady**: indicates the non-steady state. **Unknown**: unknown **NULL**: unknown

Recovery Guide:

1. Log in to the active management node and run the **su - omm** command to switch to user **omm**. Run the **${CONTROLLER_HOME}/sbin/status-oms.sh** command to check the status of OMS.
2. If floatip, okerberos, and oldap are abnormal, handle the problems by referring to ALM-12002, ALM-12004, and ALM-12005 respectively.
3. If other resources are abnormal, you are advised to view the logs of the faulty modules.

   If controller resources are abnormal, view **/var/log/Bigdata/controller/controller.log** of the faulty node.

   If CEP resources are abnormal, view **/var/log/Bigdata/omm/oms/cep/cep.log** of the faulty node.

   If AOS resources are abnormal, view **/var/log/Bigdata/controller/aos/aos.log** of the faulty node.

   If feed_watchdog resources are abnormal, view **/var/log/Bigdata/watchdog/watchdog.log** of the abnormal node.

If HTTPD resources are abnormal, view **/var/log/Bigdata/httpd/error_log** of the abnormal node.

If FMS resources are abnormal, view **/var/log/Bigdata/omm/oms/fms/fms.log** of the abnormal node.

If PMS resources are abnormal, view **/var/log/Bigdata/omm/oms/pms/pms.log** of the abnormal node.

If IAM resources are abnormal, view **/var/log/Bigdata/omm/oms/iam/iam.log** of the abnormal node.

If the GaussDB resource is abnormal, check the **/var/log/Bigdata/omm/oms/db/omm_gaussdba.log** of the abnormal node.

If NTP resources are abnormal, view **/var/log/Bigdata/omm/oms/ha/scriptlog/ha_ntp.log** of the abnormal node.

If Tomcat resources are abnormal, view **/var/log/Bigdata/tomcat/catalina.log** of the abnormal node.

4. If the fault cannot be rectified based on the logs, contact O&M personnel and send the collected fault logs.

OMA Running Status

Indicator: OMA Running Status

Description: This indicator is used to check the running status of the OMA. The status can be **Running** or **Stopped**. If the OMA is **Stopped**, the OMA is unhealthy.

Recovery Guide:

1. Log in to the unhealthy node and run the **su - omm** command to switch to user **omm**.
2. Run **${OMA_PATH}/restart_oma_app** to manually start the OMA and check again. If the check result is still unhealthy, go to **3**.
3. If manually starting the OMA cannot resolve the problem, you are advised to check the OMA logs in **/var/log/Bigdata/omm/oma/omm_agent.log**.
4. If the fault cannot be rectified based on the logs, contact O&M personnel and send the collected fault logs.

SSH Trust Between Each Node and the Active Management Node

Indicator: SSH Trust Between Each Node and the Active Management Node

Description: This indicator is used to check whether the SSH mutual trust is normal. If you can switch to another node through SSH from the active OMS node as user omm without the need of entering the password, SSH communication is normal. Otherwise, SSH communication is abnormal. In addition, if you can switch to another node through SSH from the active OMS node but fail to switch to the active OMS node from the other nodes, SSH communication is abnormal.

Recovery Guide:

1. If the indicator check result is abnormal, the SSH trust relationships between the nodes and the active management node are abnormal. In this case, check whether the permission of the **/home/omm** directory is **omm**. If non-omm users have the directory permission, the SSH trust relationship may be

abnormal. You are advised to run **chown omm:wheel** to modify the permission and check again. If the permission on the **/home/omm** directory is normal, go to **2**.

2. The SSH trust relationship exception may cause heartbeat exceptions between Controller and NodeAgent, resulting in node fault alarms. In this case, rectify the fault by referring to the handling procedure of ALM-12006.

Process Running Time

Indicator: Running Time of NodeAgent, Controller, and Tomcat

Description: This indicator is used to check the running time of the NodeAgent, Controller, and Tomcat processes. If the time is less than half an hour (1,800s), the process may have been restarted. You are advised to check the process after half an hour. If multiple check results indicate that the process runs for less than half an hour, the process is abnormal.

Recovery Guide:

1. Log in to the unhealthy node and run the **su - omm** command to switch to user **omm**.
2. Run the following command to check the PID based on the process name:

   ps -ef | grep NodeAgent
3. Run the following command to check the process startup time based on the PID:

   ps -p pid -o lstart
4. Check whether the process start time is normal. If the process restarts repeatedly, go to **5**.
5. View the related logs and analyze restart causes.

   If the runtime of NodeAgent is abnormal, check **/var/log/Bigdata/nodeagent/agentlog/agent.log**.

   If the Controller running time is abnormal, check the **/var/log/Bigdata/controller/controller.log** file.

   If the Tomcat running time is abnormal, check the **/var/log/Bigdata/tomcat/web.log** file.
6. If the fault cannot be rectified based on the logs, contact O&M personnel and send the collected fault logs.

Account and Password Expiration Check

Indicator: Account and Password Expiration Check

Description: This indicator checks the two operating system users **omm** and **ommdba** of MRS. For OS users, both the account and password expiration time must be checked. If the validity period of the account or password is not greater than 15 days, the account is abnormal.

Recovery Guide: If the validity period of the account or password is less than or equal to 15 days, contact O&M personnel.

11.7.17 Spark Health Check Indicators

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Spark service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If the indicator is abnormal, rectify the fault by referring to ALM-28001.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.18 Storm Health Check Indicators

Number of Working Nodes

Indicator: Number of Supervisors

Description: This indicator is used to check the number of available Supervisors in a cluster. If the number of available Supervisors in a cluster is less than 1, the cluster is unhealthy.

Recovery Guide: If the indicator is abnormal, go to the Streaming service instance page and click the host name of the unavailable Supervisor instance. View the host health status in the **Overview** area. If the host health status is **Good**, rectify the fault by referring to ALM-12007 Process Faults. If the status is not **Good**, rectify the fault by referring to the handling procedure of the ALM-12006 Node Faults.

Number of Idle Slots

Indicator: Number of Idle Slots

Description: This indicator is used to check the number of idle slots in a cluster. If the number of idle slots in a cluster is less than 1, the cluster is unhealthy.

Recovery Guide: If the indicator is abnormal, go to the Storm service instance page and check the health status of the Supervisor instance. If the health status of all Supervisor instances is **Good**, you need to expand the capacity of the Core node in the cluster. If not, rectify the fault by referring to ALM-12007 Process Faults.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Storm service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If the indicator is abnormal, rectify the fault by referring to the alarm "ALM-26051 Storm Service Unavailable".

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.19 Yarn Health Check Indicators

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether the Yarn service status is normal. If the number of NodeManager nodes cannot be obtained, the system is unhealthy.

Recovery Guide: If this indicator is abnormal, you can handle the alarm by referring to the alarm handling guide and make sure that the network is normal.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.7.20 ZooKeeper Health Check Indicators

Average ZooKeeper Request Processing Latency

Indicator: Average ZooKeeper Service Request Processing Latency

Description: This indicator is used to check the average delay for the ZooKeeper service to process requests. If the average delay is greater than 300 ms, the ZooKeeper service is unhealthy.

Recovery Guide: If the indicator is abnormal, check whether the network speed of the cluster is normal and whether the memory or CPU usage is too high.

ZooKeeper Connections Usage

Indicator: ZooKeeper Connections Usage

Description: This indicator is used to check whether the ZooKeeper memory usage exceeds 80%. If the disk usage exceeds the threshold, the system is unhealthy.

Recovery Guide: If the indicator is abnormal, you are advised to increase the memory available for the ZooKeeper service. The method of increasing the memory is as follows: Increase the value of **-Xmx** in the **GC_OPTS** configuration item in the ZooKeeper service. After the modification, restart the ZooKeeper service for the configuration to take effect.

Service Health Status

Indicator: Service Status

Description: This indicator is used to check whether ZooKeeper service status is normal. If the status is abnormal, the service is unhealthy.

Recovery Guide: If the indicator is abnormal, check whether the health status of the KrbServer and LdapServer services is faulty. If yes, rectify the fault. Log in to the ZooKeeper client, check whether the ZooKeeper data writing fails. If yes, find the failure cause based on the error message and handle the fault according to error message. Rectify the fault by following the procedure for handling ALM-13000.

Alarm Check

Indicator: Alarm Information

Description: This indicator is used to check whether alarms exist. If alarms exist, the service is unhealthy.

Recovery Guide: If this indicator is abnormal, you can rectify the fault by referring to the alarm handling guide.

11.8 Static Service Pool Management

11.8.1 Viewing the Status of a Static Service Pool

Scenario

MRS Manager manages and isolates service resources that are not running on YARN through the static service resource pool. It dynamically manages the total CPU, I/O, and memory resources that can be used by HDFS and YARN on the deployment node. The system supports time-based automatic adjustment of static service resource pools. This enables the cluster to automatically adjust the parameter values at different periods to ensure more efficient resource utilization.

On MRS Manager, you can view the monitoring metrics of the resources used by each service in the static service pool. The monitoring metrics are as follows:

- Service Total CPU Usage
- Service Total Disk I/O Read Speed
- Service Total Disk I/O Write Speed

- Service Total Memory Usage

Procedure

Step 1 On MRS Manager, click **System**. In the **Resource** area, click **Configure Static Service Pool**.

Step 2 Click **Status**.

Step 3 Check the system resource adjustment base values.

- **System Resource Adjustment Base** indicates the maximum volume of resources that can be used by each node in the cluster. If a node has only one service, the service exclusively occupies the available resources on the node. If a node has multiple services, all services share the available resources on the node.
- **CPU(%)** indicates the maximum number of CPUs that can be used by services on a node.
- **Memory(%)** indicates the maximum memory that can be used by services on a node.

Step 4 Check the cluster service resource usage.

In the chart area, select **All services** from the service drop-down list box. The resource usage status of all services in the service pool is displayed.

NOTE

Effective Configuration Group indicates the resource control configuration group used by the cluster service. By default, the **default** configuration group is used at all time every day, indicating that the cluster service can use all CPUs and 70% memory of the node.

Step 5 View the resource usage of a single service.

In the chart area, select a service from the service drop-down list box. The resource usage status of the service is displayed.

Step 6 You can set the interval for automatically refreshing the page.

The following refresh interval options are supported:

- **Refresh every 30 seconds**
- **Refresh every 60 seconds**
- **Stop refreshing**

Step 7 In the **Period** area, select a time range for viewing service resources. The options are as follows:

- Real time
- Last 3 hours
- Last 6 hours
- Last 24 hours
- Last week
- Last month
- Last 3 months

- Last 6 months
- Customize: If you select this option, you can customize the period for viewing monitoring data.

Step 8 Click **View** to view the service resource data in the corresponding time range.

Step 9 Customize a service resource report.

1. Click **Customize** and select the service source indicators to be displayed.
   - Service Total Disk I/O Read Speed
   - Service Total Memory Usage
   - Service Total Disk I/O Write Speed
   - Service Total CPU Usage
2. Click **OK** to save the selected monitoring metrics for display.

   NOTE

   Click **Clear** to cancel all the selected monitoring metrics in a batch.

Step 10 Export a monitoring report.

Click **Export**. MRS Manager will generate a report about the selected service resources in a specified time of period. Save the report.

NOTE

To view the curve charts of monitoring metrics in a specified period, click **View**.

----End

11.8.2 Configuring a Static Service Pool

Scenario

If you need to control the node resources that can be used by the cluster service or the CPU usage of the node used by the cluster in different time periods, you can adjust the resource base on MRS Manager and customize the resource configuration groups.

Prerequisites

- After the static service pool is configured, the HDFS and YARN services need to be restarted. During the restart, the services are unavailable.
- After a static service pool is configured, the maximum number of resources used by each service and role instance cannot exceed the upper limit.

Procedure

Step 1 Modify the system resource adjustment base.

1. On MRS Manager, click **System**. In the **Resource** area, click **Configure Static Service Pool**.
2. Click **Configuration**. The service pool configuration group management page is displayed.

3. In the **System Resource Adjustment Base** area, change the values of **CPU(%)** and **Memory(%)** .

   Modifying **System Resource Adjustment Base** limits the maximum physical CPU and memory resource percentage of nodes that can be used by the Flume, HBase, HDFS, Impala and YARN services. If multiple services are deployed on the same node, the maximum physical resource usage of all services cannot exceed the adjusted CPU or memory usage.

4. Click **Next**.

   If you need to modify the parameters again, click **Previous** in the lower part of the page.

Step 2 Modify the **default** configuration group of the service pool.

1. Click **default**. In the **Service Pool Configuration** table, set **CPU LIMIT(%)**, **CPU SHARE(%)**, **I/O(%)**, and **Memory(%)** for the Flume, HBase, HDFS, Impala and YARN services.

   NOTE

   - The sum of **CPU LIMIT(%)** used by all services can exceed 100%.
   - The sum of **CPU SHARE(%)** and **I/O(%)** used by all services must be 100%. For example, if CPU resources are allocated to the HDFS and Yarn services, the total CPU resources allocated to the two services are 100%.
   - The sum of **Memory(%)** used by all services can be greater than, smaller than, or equal to 100%.
   - **Memory(%)** cannot take effect dynamically and can only be modified in the default configuration group.

2. Click in the blank area of the page to complete the editing. MRS Manager generates the correct values of service pool parameters in the **Detailed Configuration** area based on the cluster hardware resources and allocation information.

3. You can click  on the right of **Detailed Configuration** to modify the parameter values of the service pool based on service requirements.

   In the **Service Pool Configuration** area, click the specified service name. The **Detailed Configuration** area displays only the parameters of the service. Manual changing of parameter values does not refresh the service resource usage. In added configuration groups, the configuration group numbers of the parameters that take effect dynamically will be displayed. For example, **HBase: RegionServer: dynamic-config1.RES_CPUSET_PERCENTAGE**. The parameter functions do not change.

   Table 11-29 Parameters of the static service pool

Parameter	Description
– RES_CPUSET_PERCENTAGE – dynamic-configX.RES_CPUSET_PERCENTAGE	Configures the service CPU percentage.

Parameter	Description
– RES_CPU_SHARE – dynamic-configX.RES_CPU_SHARE	Configures the service CPU share.
– RES_BLKIO_WEIGHT – dynamic-configX.RES_BLKIO_WEIGHT	Configures service I/O usage.
HBASE_HEAPSIZE	Configures the maximum JVM memory for RegionServer.
HADOOP_HEAPSIZE	Configures the maximum JVM memory of a DataNode.
yarn.nodemanager.resource.memory-mb	Configures the memory that can be used by NodeManager on the current node.
dfs.datanode.max.locked.memory	Configures the maximum memory that can be used by a DataNode as the HDFS cache.
FLUME_HEAPSIZE	Configures the maximum JVM memory that can be used by each Flume instance.
IMPALAD_MEM_LIMIT	Configures the maximum memory that can be used by an Impalad instance.

Step 3 Add a customized resource configuration group.

1. Determine whether to automatically adjust resource configurations based on the time.

   If yes, go to **Step 3.2**.

   If no, go to **Step 4**.

2. Click  to add a resource configuration group. In the **Scheduling Time** area, click  . The time policy configuration page is displayed.

   Modify the following parameters based on service requirements and click **OK**.

   – **Repeat**: If selected, the resource configuration group runs repeatedly based on the scheduling period. If not selected, set the date and time when the configuration of the group of resources can be applied.

   – **Repeat Policy**: can be set to **Daily**, **Weekly**, and **Monthly**. This parameter is valid only when **Repeat** is selected.

   – **Between**: indicates the time period between the start time and end time when the resource configuration is applied. Set a unique time range. If the time range overlaps with that of an existing group of resource configuration, the time range cannot be saved. This parameter is valid only when **Repeat** is selected.

NOTE

- The **default** group of resource configuration takes effect in all undefined time segments.
- The newly added resource group is a parameter set that takes effect dynamically in a specified time range.
- The newly added resource group can be deleted. A maximum of four resource configuration groups that take effect dynamically can be added.
- Select a repetition policy. If the end time is earlier than the start time, the next day is labeled by default. For example, if a validity period ranges from 22:00 to 06:00, the customized resource configuration takes effect from 22:00 on the current day to 06:00 on the next day.
- If the repeat policy types of multiple configuration groups are different, the time ranges can overlap. The policy types are listed as follows by priority from low to high: daily, weekly, and monthly. The following is an example. There are two resource configuration groups using the monthly and daily policies, respectively. Their application time ranges in a day overlap as follows: [04:00 to 07:00] and [06:00 to 08:00]. In this case, the configuration of the group that uses the monthly policy prevails.
- If the repeat policy types of multiple resource configuration groups are the same, the time ranges of different dates can overlap. For example, if there are two weekly scheduling groups, you can set the same time range on different day for them, such as to 04:00 to 07:00, on Monday and Wednesday, respectively.

3. On the **Service Pool Configuration** page, modify the resource configuration of each service. Click the blank area on the page to complete the editing, and go to **Step 4**.

   You can click on the right of **Service Pool Configuration** to modify the parameters. Click in the **Detailed Configuration** area to manually update the parameter values generated by the system based on service requirements.

Step 4 Saves the settings.

Click **Save**. In the **Save Configuration** dialog box, select **Restart the affected services or instances**. Click **OK** to save the settings and restart related services.

Operation succeeded is displayed. click **Finish**. The service is started successfully.

----End

11.9 Tenant Management

11.9.1 Overview

Definition

An MRS cluster provides various resources and services for multiple organizations, departments, or applications to share. The cluster provides tenants as a logical entity to use these resources and services. A mode involving different tenants is called multi-tenant mode. Currently, only the analysis cluster supports tenant management.

Principles

The MRS cluster provides the multi-tenant function. It supports a layered tenant model and allows dynamic adding or deleting of tenants to isolate resources. It dynamically manages and configures tenants' computing and storage resources.

The computing resources indicate tenants' Yarn task queue resources. The task queue quota can be modified, and the task queue usage status and statistics can be viewed.

The storage resources can be stored on HDFS. You can add and delete the HDFS storage directories of tenants, and set the quotas of file quantity and the storage space of the directories.

As the unified tenant management platform of MRS clusters, MRS Manager provides enterprises with time-tested multi-tenant management models, enabling centralized tenant and service management. Tenants can create and manage tenants in a cluster based on service requirements.

- Roles, computing resources, and storage resources are automatically created when tenants are created. By default, all permissions of the new computing resources and storage resources are allocated to a tenant's roles.
- Permissions to view the current tenant's resources, add a subtenant, and manage the subtenant's resources are granted to the tenant's roles by default.
- After you have modified the tenant's computing or storage resources, permissions of the tenant's roles are automatically updated.

MRS Manager supports a maximum of 512 tenants. The tenants that are created by default in the system contain **default**. Tenants that are in the topmost layer with the default tenant are called level-1 tenants.

Resource Pools

Yarn task queues support only the label-based scheduling policy. This policy enables Yarn task queues to associate NodeManagers that have specific node labels. In this way, Yarn tasks run on specified nodes so that tasks are scheduled and certain hardware resources are utilized. For example, Yarn tasks requiring a large memory capacity can run on nodes with a large memory capacity by means of label association, preventing poor service performance.

In an MRS cluster, the tenant logically divides Yarn cluster nodes to combine multiple NodeManagers into a resource pool. Yarn task queues can be associated with specified resource pools by configuring queue capacity policies, ensuring efficient and independent resource utilization in the resource pools.

MRS Manager supports a maximum of 50 resource pools. The system has a **Default** resource pool.

11.9.2 Creating a Tenant

Scenario

You can create a tenant on MRS Manager to specify the resource usage.

Prerequisites

- A tenant name has been planned. The name must not be the same as that of a role or Yarn queue that exists in the current cluster.
- If a tenant requires storage resources, a storage directory has been planned based on service requirements, and the planned directory does not exist under the HDFS directory.
- The resources that can be allocated to the current tenant have been planned and the sum of the resource percentages of direct sub-tenants under the parent tenant at every level does not exceed 100%.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click **Create Tenant**. On the page that is displayed, configure tenant properties.

Table 11-30 Tenant parameters

Parameter	Description
Name	Specifies the name of the current tenant. The value consists of 3 to 20 characters, and can contain letters, digits, and underscores (_).
Tenant Type	The options include **Leaf** and **Non-leaf**. If **Leaf** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. If **Non-leaf** is selected, sub-tenants can be added to the current tenant.
Dynamic Resources	Specifies the dynamic computing resources for the current tenant. The system automatically creates a task queue named after the tenant name in Yarn. When dynamic resources are not **Yarn**, the system does not automatically create a task queue.
Default Resource Pool Capacity (%)	Specifies the percentage of the computing resources used by the current tenant in the **default** resource pool.
Default Resource Pool Max. Capacity (%)	Specifies the maximum percentage of the computing resources used by the current tenant in the **default** resource pool.
Storage Resource	Specifies storage resources for the current tenant. The system automatically creates a file folder named after the tenant name in the **/tenant** directory. When a tenant is created for the first time, the system automatically creates the **/tenant** directory in the HDFS root directory. If storage resources are not **HDFS**, the system does not create a storage directory under the root directory of HDFS.

Parameter	Description
Space Quota (MB)	Specifies the quota for HDFS storage space used by the current tenant. The value ranges from **1** to **8796093022208**. The unit is MB. This parameter indicates the maximum HDFS storage space that can be used by a tenant, but does not indicate the actual space used. If the value is greater than the size of the HDFS physical disk, the maximum space available is the full space of the oHDFS physical disk. **NOTE** To ensure data reliability, one copy of a file is automatically generated when the file is stored in HDFS. That is, two copies of the same file are stored by default. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value of **Storage Space Quota** is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).
Storage Path	Specifies the tenant's HDFS storage directory. The system automatically creates a file folder named after the tenant name in the **/tenant** directory by default. For example, the default HDFS storage directory for tenant **ta1** is **tenant/ta1**. When a tenant is created for the first time, the system automatically creates the **/tenant** directory in the HDFS root directory. The storage path is customizable.
Service	Specifies other service resources associated with the current tenant. HBase is supported. To configure this parameter, click **Associate Services**. In the dialog box that is displayed, set **Service** to **HBase**. If **Association Mode** is set to **Exclusive**, service resources are occupied exclusively. If **share** is selected, service resources are shared.
Description	Specifies the description of the current tenant.

Step 3 Click **OK** to save the settings.

It takes a few minutes to save the settings. If the **Tenant created successfully** is displayed in the upper-right corner, the tenant is added successfully.

NOTE

- Roles, computing resources, and storage resources are automatically created when tenants are created.
- The new role has permissions on the computing and storage resources. The role and its permissions are controlled by the system automatically and cannot be controlled manually under **Manage Role**.
- If you want to use the tenant, create a system user and assign the Manager_tenant role and the role corresponding to the tenant to the user. For details, see **Creating a User**.

----End

Related Tasks

Viewing an added tenant

Step 1 On MRS Manager, click **Tenant**.

Step 2 In the tenant list on the left, click the name of the added tenant.

The **Summary** tab is displayed on the right by default.

Step 3 View **Basic Information**, **Resource Quota**, and **Statistics** of the tenant.

If HDFS is in the **Stopped** state, **Available** and **Used** of **Space** in **Resource Quota** are **unknown.**

----End

11.9.3 Creating a Sub-tenant

Scenario

You can create a sub-tenant on MRS Manager if the resources of the current tenant need to be further allocated.

Prerequisites

- A parent tenant has been added.
- A tenant name has been planned. The name must not be the same as that of a role or Yarn queue that exists in the current cluster.
- If a sub-tenant requires storage resources, a storage directory has been planned based on service requirements, and the planned directory does not exist under the storage directory of the parent tenant.
- The resources that can be allocated to the current tenant have been planned and the sum of the resource percentages of direct sub-tenants under the parent tenant at every level does not exceed 100%.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 In the tenant list on the left, move the cursor to the tenant node to which a sub-tenant is to be added. Click **Create sub-tenant**. On the displayed page, configure the sub-tenant attributes according to the following table:

Table 11-31 Sub-tenant parameters

Parameter	Description
Parent tenant	Specifies the name of the parent tenant.
Name	Specifies the name of the current tenant. The value consists of 3 to 20 characters, and can contain letters, digits, and underscores (_).

Parameter	Description
Tenant Type	The options include **Leaf** and **Non-leaf**. If **Leaf** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. If **Non-leaf** is selected, sub-tenants can be added to the current tenant.
Dynamic Resources	Specifies the dynamic computing resources for the current tenant. The system automatically creates a task queue named after the sub-tenant name in the Yarn parent queue. When dynamic resources are not **Yarn**, the system does not automatically create a task queue. If the parent tenant does not have dynamic resources, the sub-tenant cannot use dynamic resources.
Default Resource Pool Capacity (%)	Specifies the percentage of the resources used by the current tenant. The base value is the total resources of the parent tenant.
Default Resource Pool Max. Capacity (%)	Specifies the maximum percentage of the computing resources used by the current tenant. The base value is the total resources of the parent tenant.
Storage Resource	Specifies storage resources for the current tenant. The system automatically creates a file in the HDFS parent tenant directory. The file is named the same as the name of the sub-tenant. If storage resources are not **HDFS**, the system does not create a storage directory under the root directory of HDFS. If the parent tenant does not have storage resources, the sub-tenant cannot use storage resources.
Space Quota (MB)	Specifies the quota for HDFS storage space used by the current tenant. The minimum value is 1, and the maximum value is the total storage quota of the parent tenant. The unit is MB. This parameter indicates the maximum HDFS storage space that can be used by a tenant, but does not indicate the actual space used. If the value is greater than the size of the HDFS physical disk, the maximum space available is the full space of the oHDFS physical disk. If the quota is greater than the quota of the parent tenant, the actual storage capacity is subject to the quota of the parent tenant. **NOTE** To ensure data reliability, one copy of a file is automatically generated when the file is stored in HDFS. That is, two copies of the same file are stored by default. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).

Parameter	Description
Storage Path	Specifies the tenant's HDFS storage directory. The system automatically creates a file folder named after the sub-tenant name in the directory of the parent tenant by default. For example, if the sub-tenant is **ta1s** and the parent directory is **tenant/ta1**, the system sets this parameter for the sub-tenant to **tenant/ta1/ta1s**. The storage path is customizable in the parent directory. The parent directory for the storage path must be the storage directory of the parent tenant.
Service	Specifies other service resources associated with the current tenant. HBase is supported. To configure this parameter, click **Associate Services**. In the dialog box that is displayed, set **Service** to **HBase**. If **Association Mode** is set to **Exclusive**, service resources are occupied exclusively. If **share** is selected, service resources are shared.
Description	Specifies the description of the current tenant.

Step 3 Click **OK** to save the settings.

It takes a few minutes to save the settings. If the **Tenant created successfully** is displayed in the upper-right corner, the tenant is added successfully. The tenant is created successfully.

NOTE

- Roles, computing resources, and storage resources are automatically created when tenants are created.
- The new role has permissions on the computing and storage resources. The role and its permissions are controlled by the system automatically and cannot be controlled manually under **Manage Role**.
- When using this tenant, create a system user and assign the user a related tenant role. For details, see **Creating a User**.

----End

11.9.4 Deleting a tenant

Scenario

You can delete a tenant that is not required on MRS Manager.

Prerequisites

- A tenant has been added.
- You have checked whether the tenant to be deleted has sub-tenants. If the tenant has sub-tenants, delete them; otherwise, you cannot delete the tenant.

- The role of the tenant to be deleted cannot be associated with any user or user group. For details about how to cancel the binding between a role and a user, see **Modifying User Information**.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 In the tenant list on the left, move the cursor to the tenant node to be deleted and click **Delete**.

The **Delete Tenant** dialog box is displayed. If you want to save the tenant data, select **Reserve the data of this tenant**. Otherwise, the tenant's storage space will be deleted.

Step 3 Click OK to save the settings.

It takes a few minutes to save the configuration. After the tenant is deleted successfully, the role and storage space of the tenant are also deleted.

NOTE

- After the tenant is deleted, the task queue of the tenant still exists in Yarn.
- If you choose not to reserve data when deleting the parent tenant, data of sub-tenants is also deleted if the sub-tenants use storage resources.

----End

11.9.5 Managing a Tenant Directory

Scenario

You can manage the HDFS storage directory used by a specific tenant on MRS Manager. The management operations include adding a tenant directory, modifying the directory file quota, modifying the storage space, and deleting a directory.

Prerequisites

A tenant associated with HDFS storage resources has been added.

Procedure

- Viewing a tenant directory
  a. On MRS Manager, click **Tenant**.
  b. In the tenant list on the left, click the target tenant.
  c. Click the **Resource** tab.
  d. View the **HDFS Storage** table.
     - The Quota column indicates the quantity quotas of files and directories.
     - The **Storage Space Quota** column indicates the storage space size of the tenant directory.

- Adding a tenant directory
  a. On MRS Manager, click **Tenant**.
  b. In the tenant list on the left, click the tenant whose HDFS storage directory needs to be added.
  c. Click the **Resource** tab.
  d. In the **HDFS Storage** table, click **Create Directory**.
     - In **Parent Directory**, select a storage directory of a parent tenant.
       This parameter applies only to sub-tenants. If the parent tenant has multiple directories, select any of them.
     - Set **Path** to a tenant directory path.
       NOTE
       - If the current tenant is not a sub-tenant, the new path is created in the HDFS root directory.
       - If the current tenant is a sub-tenant, the new path is created in the specified directory.

       A complete HDFS storage directory can contain a maximum of 1,023 characters. An HDFS directory name contains digits, letters, spaces, and underscores (_). The name cannot start or end with a space.
     - Set **Quota** to the quotas of file and directory quantity.
       Maximum Number of Files/Directories is optional. Its value ranges from **1** to **9223372036854775806**.
     - Set **Storage Space Quota** to the storage space size of the tenant directory.
       The value of **Storage Space Quota** ranges from **1** to **8796093022208**.
       NOTE
       To ensure data reliability, one copy of a file is automatically generated when the file is stored in HDFS. That is, two copies of the same file are stored by default. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value of **Storage Space Quota** is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).
  e. Click **OK**. The system creates tenant directories in the HDFS root directory.
- Modify a tenant directory.
  a. On MRS Manager, click **Tenant**.
  b. In the tenant list on the left, click the tenant whose HDFS storage directory needs to be modified.
  c. Click the **Resource** tab.
  d. In the **HDFS Storage** table, click **Modify** in the **Operation** column of the specified tenant directory.
     - Set **Quota** to the quotas of file and directory quantity.
       Maximum Number of Files/Directories is optional. Its value ranges from **1** to **9223372036854775806**.

- Set **Storage Space Quota** to the storage space size of the tenant directory.

  The value of **Storage Space Quota** ranges from **1** to **8796093022208**.

  NOTE

  To ensure data reliability, one copy of a file is automatically generated when the file is stored in HDFS. That is, two copies of the same file are stored by default. The HDFS storage space indicates the total disk space occupied by all these copies. For example, if the value of **Storage Space Quota** is set to **500**, the actual space for storing files is about 250 MB (500/2 = 250).

e. Click **OK**.

- Delete a tenant directory.

  a. On MRS Manager, click **Tenant**.

  b. In the tenant list on the left, click the tenant whose HDFS storage directory needs to be deleted.

  c. Click the **Resource** tab.

  d. In the **HDFS Storage** table, click **Delete** in the **Operation** column of the specified tenant directory.

  The default HDFS storage directory set during tenant creation cannot be deleted. Only the newly added HDFS storage directory can be deleted.

  e. Click **OK**.

11.9.6 Restoring Tenant Data

Scenario

Tenant data is stored on Manager and in cluster components by default. When components are restored from faults or reinstalled, some tenant configuration data may be abnormal. In this case, you can manually restore the tenant data.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 In the tenant list on the left, click a tenant node.

Step 3 Check the status of the tenant data.

1. In **Summary**, check the color of the circle on the left of **Basic Information**. Green indicates that the tenant is available and gray indicates that the tenant is unavailable.
2. Click **Resources** and check the circle on the left of **Yarn** or **HDFS Storage**. Green indicates that the resource is available, and gray indicates that the resource is unavailable.
3. Click **Service Association** and check the **Status** column of the associated service table. **Good** indicates that the component can provide services for the associated tenant. **Bad** indicates that the component cannot provide services for the tenant.

4. If any check result is abnormal, go to **Step 4** to restore tenant data.

Step 4 Click **Restore Tenant Data**.

Step 5 In the **Restore Tenant Data** window, select one or more components whose data needs to be restored. Click **OK**. The system automatically restores the tenant data.

----End

11.9.7 Creating a Resource Pool

Scenario

In an MRS cluster, users can logically divide Yarn cluster nodes to combine multiple NodeManagers into a Yarn resource pool. Each NodeManager belongs to one resource pool only. The system contains a **Default** resource pool by default. All NodeManagers that are not added to customized resource pools belong to this resource pool.

You can create a customized resource pool on MRS Manager and add hosts that have not been added to other customized resource pools to it.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click the **Resource Pools** tab.

Step 3 Click **Add Resource Pool**.

Step 4 In **Create Resource Pool**, set the properties of the resource pool.

- **Name**: Enter a name for the resource pool. The name of the newly created resource pool cannot be **Default**.

  The name consists of 1 to 20 characters and can contain digits, letters, and underscores (_) but cannot start with an underscore (_).

- **Hosts**: In the host list on the left, select the name of a specified host and click 

to add the selected host to the resource pool. Only hosts in the cluster can be selected. The host list of a resource pool can be left blank.

Step 5 Click **OK**.

Step 6 After a resource pool is created, users can view the **Name**, **Members**, **Type**, **vCore** and **Memory** in the resource pool list. Hosts that are added to the customized resource pool are no longer members of the **Default** resource pool.

----End

11.9.8 Modifying a Resource Pool

Scenario

You can modify members of an existing resource pool on MRS Manager.

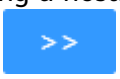
Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click the **Resource Pools** tab.

Step 3 Locate the row that contains the specified resource pool, and click **Modify** in the **Operation** column.

Step 4 In **Modify Resource Pool**, modify **Added Hosts**.

- Adding a host: Select the name of a specified host in host list on the left and click 

to add the selected host to the resource pool.
- Deleting a host: In the host list on the right, select the name of a specified host and click 

to add the selected host to the resource pool. The host list of a resource pool can be left blank.

Step 5 Click **OK**.

----End

11.9.9 Deleting a Resource Pool

Scenario

You can delete an existing resource pool on MRS Manager.

Prerequisites

- Any queue in a cluster cannot use the resource pool to be deleted as the default resource pool. Before deleting the resource pool, cancel the default resource pool. For details, see **Configuring a Queue**.
- Resource distribution policies of all queues have been cleared from the resource pool being deleted. For details, see **Clearing Configuration of a Queue**.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click the **Resource Pools** tab.

Step 3 Locate the row that contains the specified resource pool, and click **Delete** in the **Operation** column.

In the displayed dialog box, click **OK**.

----End

11.9.10 Configuring a Queue

Scenario

This section describes how to modify the queue configuration for a specified tenant on MRS Manager.

Prerequisites

A tenant associated with Yarn and allocated dynamic resources has been added.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click the **Dynamic Resource Plan** tab.

Step 3 Click the **Queue Configuration** tab.

Step 4 In the tenant queue table, click **Modify** in the **Operation** column of the specified tenant queue.

NOTE

In the tenant list on the left of the **Tenant Management** tab, click the target tenant. In the window that is displayed, choose **Resource**. On the page that is displayed, click to open the queue modification page.

Table 11-32 Queue configuration parameters

Parameter	Description
Maximum Application	Specifies the maximum number of applications. The value ranges from 1 to 2147483647.
Maximum AM Resource Percent	Specifies the maximum percentage of resources that can be used to run the ApplicationMaster in a cluster. The value ranges from 0 to 1.
Minimum User Limit Percent (%)	Specifies the minimum percentage of resources consumed by a user. The value ranges from 0 to 100.
User Limit Factor	Specifies the limit factor of the maximum user resource usage. The maximum user resource usage percentage can be obtained by multiplying the limit factor with the percentage of the tenant's actual resource usage in the cluster. The minimum value is **0**.
Status	Specifies the current status of a resource plan. The values are **Running** and **Stopped**.

Parameter	Description
Default Resource Pool	Specifies the resource pool used by a queue. The default value is **Default**. If you want to change the resource pool, configure the queue capacity first. For details, see **Configuring the Queue Capacity Policy of a Resource Pool**.

----End

11.9.11 Configuring the Queue Capacity Policy of a Resource Pool

Scenario

After a resource pool is added, the capacity policies of available resources need to be configured for Yarn task queues. This ensures that tasks in the resource pool are running properly. Each queue can be configured with the queue capacity policy of only one resource pool. Users can view the queues in any resource pool and configure queue capacity policies. After the queue policies are configured, Yarn task queues and resource pools are associated.

You can configure queue policies on MRS Manager.

Prerequisites

- A resource pool has been added.
- The task queues are not associated with other resource pools. By default, all queues are associated with the **Default** resource pool.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click the **Dynamic Resource Plan** tab.

Step 3 In **Resource Pools**, select a specified resource pool.

Available Resource Quota: indicates that all resources in each resource pool are available for queues by default.

Step 4 Locate the specified queue in the **Resource Allocation** table, and click **Modify** in the **Operation** column.

Step 5 In **Modify Resource Allocation**, configure the resource capacity policy of the task queue in the resource pool.

- **Capacity (%)**: specifies the percentage of the current tenant's computing resource usage.
- **Maximum Capacity (%)**: specifies the percentage of the current tenant's maximum computing resource usage.

Step 6 Click **OK** to save the settings.

----End

11.9.12 Clearing Configuration of a Queue

Scenario

Users can clear the configuration of a queue on MRS Manager when the queue does not need resources from a resource pool or if a resource pool needs to be disassociated from the queue. Clearing queue configurations means that the resource capacity policy of the queue is canceled.

Prerequisites

If a queue is to be unbound from a resource pool, this resource pool cannot serve as the default resource pool of the queue. Therefore, you must first change the default resource pool of the queue to another one. For details, see **Configuring a Queue**.

Procedure

Step 1 On MRS Manager, click **Tenant**.

Step 2 Click the **Dynamic Resource Plan** tab.

Step 3 In **Resource Pools**, select a specified resource pool.

Step 4 Locate the specified queue in the **Resource Allocation** table, and click **Clear** in the **Operation** column.

In the **Clear Queue Configuration** dialog box, click **OK** to clear the queue configuration in the current resource pool.

NOTE

If no resource capacity policy is configured for a queue, the clearing function is unavailable for the queue by default.

----End

11.10 Backup and Restoration

11.10.1 Introduction

Purpose

MRS Manager provides backup and restoration for user data and system data. The backup function is provided based on components to back up Manager data (including OMS data and LdapServer data), Hive user data, component metadata saved in DBService, and HDFS metadata.

Backup and restoration tasks are performed in the following scenarios:

- Routine backup is performed to ensure the data security of the system and components.
- If the system is faulty, the data backup can be used to recover the system.
- If the active cluster is completely faulty, a mirror cluster identical to the active cluster needs to be created. You can use the backup data to restore the active cluster.

Table 11-33 Backing up metadata

Backup Type	Backup Content
OMS	Database data (excluding alarm data) and configuration data in the cluster management system to be backed up by default
LdapServer	User information, including the username, password, key, password policy, and group information
DBService	Metadata of the components (Hive) managed by DBService
NameNode	HDFS metadata.

Principles

Task

Before backup or restoration, you need to create a backup or restoration task and set task parameters, such as the task name, backup data source, and type of backup file save path. Data backup and restoration can be performed by executing backup and restoration tasks. When the Manager is used to recover the data of HDFS, HBase, Hive, and NameNode, no cluster can be accessed.

Each backup task can back up data of different data sources and generates an independent backup file for each data source. All the backup files generated in each backup task form a backup file set, which can be used in restoration tasks. Backup data can be stored on Linux local disks, local cluster HDFS, and standby cluster HDFS. The backup task provides the full backup or incremental backup policies. HDFS and Hive backup tasks support the incremental backup policy, while OMS, LdapServer, DBService, and NameNode backup tasks support only the full backup policy.

NOTE

Task execution rules:

- If a task is being executed, the task cannot be executed repeatedly and other tasks cannot be started at the same time.
- The interval at which a periodical task is automatically executed must be greater than 120s; otherwise, the task is postponed and will be executed in the next period. Manual tasks can be executed at any interval.
- When a periodic task is to be automatically executed, the current time cannot be 120s later than the task start time; otherwise, the task is postponed and executed in the next period.
- When a periodic task is locked, it cannot be automatically executed and needs to be manually unlocked.
- Before an OMS, LdapServer, DBService, or NameNode backup task starts, ensure that the LocalBackup partition on the active management node has more than 20 GB available space. Otherwise, the backup task cannot be started.
- When you are planning backup and restoration tasks, select the data to be backed up or restored strictly based on the service logic, data store structure, and database or table association. The system creates a default periodic backup task **default** whose execution interval is 24 hours to perform full backup of OMS, LdapServer, DBService, and NameNode data to the Linux local disk.

Specifications

Table 11-34 Backup and restoration feature specifications

Item	Specifications
Maximum number of backup or restoration tasks	100
Number of concurrent running tasks	1
Maximum number of waiting tasks	199
Maximum size of backup files on a Linux local disk (GB)	600

Table 11-35 Specifications of the **default** task

Item	OMS	LdapServer	DBService	NameNode
Backup period	1 hour			
Maximum number of copies	2			
Maximum size of a backup file	10 MB	20 MB	100 MB	1.5 GB

Item	OMS	LdapServer	DBService	NameNode
Maximum size of disk space used	20 MB	40 MB	200 MB	3 GB
Save path of backup data	*Data save path*/**LocalBackup/** of the active and standby management nodes			

NOTE

The backup data of the **default** task must be periodically transferred and saved outside the cluster based on the enterprise O&M requirements.

11.10.2 Backing Up Metadata

Scenario

To ensure the security of metadata either on a routine basis or before and after performing critical metadata operations (such as scale-out, scale-in, patch installation, upgrades, and migration), metadata must be backed up. The backup data can be used to recover the system if an exception occurs or if the operation has not achieved the expected result. This minimizes the adverse impact on services. Metadata includes data of OMS, LdapServer, DBService, and NameNode. MRS Manager data to be backed up includes OMS data and LdapServer data.

By default, metadata backup is supported by the **default** task. This section describes how to create a backup task and back up metadata on MRS Manager. Both automatic backup tasks and manual backup tasks are supported.

Prerequisites

- A standby cluster for backing up data has been created, and the network is connected. The inbound rules of the two security groups on the peer cluster have been added to the two security groups in each cluster to allow all access requests of all protocols and ports of all ECSs in the security groups.
- The backup type, period, policy, and other specifications have been planned based on the service requirements and you have checked whether *Data storage path*/**LocalBackup/** has sufficient space on the active and standby management nodes.

Procedure

Step 1 Create a backup task.

1. On MRS Manager, choose **System** > **Back Up Data**.
2. Click **Create Backup Task**.

Step 2 Configure a backup policy.

1. Set **Task Name** to the name of the backup task.

2. Set **Backup Mode** to the type of the backup task. **Periodic** indicates that the backup task is periodically executed. **Manual** indicates that the backup task is manually executed.

   To create a periodic backup task, set the following parameters:

   - **Started**: indicates the time when the task is started for the first time.
   - **Period**: indicates the task execution interval. The options include **By hour** and **By day**.
   - **Backup Policy**: indicates the volume of data to be backed up in each task execution. The options include **Full backup at the first time and incremental backup later**, **Full backup every time**, and **Full backup once every n times**. If you select **Full backup once every n times**, you need to specify the value of **n**.

Step 3 Select backup sources.

In the **Configuration** area, select **OMS** and **LdapServer** under **Metadata**.

Step 4 Set backup parameters.

1. Set **Path Type** of **OMS** and **LdapServer** to a backup directory type.

   The following backup directory types are supported:

   - **LocalDir**: indicates that the backup files are stored on the local disk of the active management node and the standby management node automatically synchronizes the backup files. By default, the backup files are stored in *Data storage path***/LocalBackup/**. If you select **LocalDir**, you need to set the maximum number of copies to specify the number of backup files that can be retained in the backup directory.
   - **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster. If you select **SFTP**, set the following parameters:

     - **Target Path**: indicates the HDFS directory for storing the backup files. The save path cannot be an HDFS hidden directory, such as a snapshot or recycle bin directory, or a default system directory.
     - **Max Number of Backup Copies**: indicates the number of backup files that can be retained in the backup directory.
     - **Target Instance Name**: indicates the NameService name of the backup directory. The default value is **hacluster**.

2. Click **OK**.

Step 5 Execute the backup task.

In the **Operation** column of the created task in the backup task list, click **Back Up Now** if **Backup Mode** is set to **Periodic** or click **Start** If **Backup Mode** is set to **Manual** to execute the backup task.

After the backup task is executed, the system automatically creates a subdirectory for each backup task in the backup directory. The format of the subdirectory name is *Backup task name_Task creation time*, and the subdirectory is used to save data source backup files. The format of the backup file name is *Version_Data source_Task execution time*.**tar.gz**.

----End

11.10.3 Restoring Metadata

Scenario

You need to restore metadata in the following scenarios: A user modifies or deletes data unexpectedly, data needs to be retrieved, system data becomes abnormal or does not achieve the expected result, all modules are faulty, and data is migrated to a new cluster.

This section describes how to restore metadata on MRS Manager. Only manual restoration tasks are supported.

NOTICE

- Data restoration can be performed only when the system version is consistent with that during data backup.
- To restore the data when services are normal, manually back up the latest management data first and then restore the data. Otherwise, the data that is generated after the data backup and before the data restoration will be lost.
- Use the OMS data and LdapServer data backed up at the same time to restore data. Otherwise, the service and operation may fail.
- By default, MRS clusters use DBService to store Hive metadata.

Impact on the System

- After the data is restored, the data generated between the backup time and restoration time is lost.
- After the data is restored, the configuration of the components that depend on DBService may expire and these components need to be restarted.

Prerequisites

- The data in the OMS and LdapServer backup files has been backed up at the same time.
- The status of the OMS resources and the LdapServer instances is normal. If the status is abnormal, data restoration cannot be performed.
- The status of the cluster hosts and services is normal. If the status is abnormal, data restoration cannot be performed.
- The cluster host topologies during data restoration and data backup are the same. If the topologies are different, data restoration cannot be performed and you need to back up data again.
- The services added to the cluster during data restoration and data backup are the same. If the services are different, data restoration cannot be performed and you need to back up data again.
- The status of the active and standby DBService instances is normal. If the status is abnormal, data restoration cannot be performed.
- The upper-layer applications depending on the MRS cluster have been stopped.

- On MRS Manager, you have stopped all the NameNode role instances whose data is to be recovered. Other HDFS role instances are running properly. After data is recovered, the NameNode role instances need to be restarted and cannot be accessed before the restart.
- You have checked whether NameNode backup files have been stored in the *Data save path*/**LocalBackup/** directory on the active management node.

Procedure

Step 1 Check the location of backup data.

1. On MRS Manager, choose **System** > **Back Up Data**.
2. In the row where the specified backup task resides, choose **More** > **View History** in the **Operation** column to display the historical execution records of the backup task. In the window that is displayed, select a success record and click **View Backup Path** in the corresponding column to view its backup path information. Find the following information:
   - **Backup Object**: indicates the backup data source.
   - **Backup Path**: indicates the full path where backup files are stored.
3. Select the correct path, and manually copy the full path of backup files in **Backup Path**.

Step 2 Create a restoration task.

1. On MRS Manager, choose System > Recovery Management.
2. On the page that is displayed, click **Create Restoration Task**.
3. Set **Task Name** to the name of the restoration task.

Step 3 Select restoration sources.

In **Configuration**, select the metadata component whose data is to be restored.

Step 4 Set the restoration parameters.

1. Set **Path Type** to a backup directory type.
2. The settings vary according to backup directory types:
   - **LocalDir**: indicates that the backup files are stored on the local disk of the active management node. If you select **LocalDir**, you need to set **Source Path** to specify the full path of the backup file. For example, *Data storage path*/**LocalBackup/***Backup task name_Task creation time*/*Data source_Task execution time*/*Version number_Data source_Task execution time*.**tar.gz**.
   - **LocalHDFS**: indicates that the backup files are stored in the HDFS directory of the current cluster. If you select **SFTP**, set the following parameters:
     - **Source Path**: indicates the full HDFS path of a backup file. for example, *Backup path/Backup task name_Task creation time/Version_Data source_Task execution time*.**tar.gz**.
     - **Source Instance Name**: indicates the name of NameService corresponding to the backup directory when a restoration task is being executed. The default value is **hacluster**.

3. Click **OK**.

Step 5 Execute the restoration task.

In the restoration task list, locate the row where the created task resides, and click **Start** in the **Operation** column.

- After the restoration is successful, the progress bar is in green.
- After the restoration is successful, the restoration task cannot be executed again.
- If the restoration task fails during the first execution, rectify the fault and try to execute the task again by clicking **Start**.

Step 6 Determine what metadata has been restored.

- If the OMS and LdapServer metadata is restored, go to **Step 7**.
- If DBService data is restored, no further action is required.
- Restore NameNode data. On MRS Manager, choose **Services > HDFS > More > Restart Service**. The task is complete.

Step 7 Restarting Manager for the recovered data to take effect

1. In MRS Manager, Choose **LdapServer** > **More** > **Restart Service** and click **OK**. Wait until the LdapServer service is restarted successfully.
2. Log in to the active management node. For details, see **Determining Active and Standby Management Nodes of Manager**.
3. Run the following command to restart OMS:

   sh ${BIGDATA_HOME}/om-0.0.1/sbin/restart-oms.sh

   The command has been executed successfully if the following information is displayed:

   ```
   start HA successfully.
   ```

4. On MRS Manager, choose **KrbServer > More > Synchronize Configuration**. Do not select Restart the services and instances whose configuration has expired. Click **OK** and wait until the KrbServer service configuration is synchronized and restarted successfully.
5. Choose **Services > More > Synchronize Configuration**. Do not select Restart the services and instances whose configuration has expired. Click **OK** and wait until the cluster is configured and synchronized successfully.
6. Choose **Services > More > Stop Cluster**. After the cluster is stopped, choose **Services > More> Start Cluster**.

----End

11.10.4 Modifying a Backup Task

Scenario

This section describes how to modify the parameters of a created backup task on MRS Manager to meet changing service requirements. The parameters of restoration tasks can be viewed but not modified.

Impact on the System

After a backup task is modified, the new parameters take effect when the task is executed next time.

Prerequisites

- A backup task has been created.
- A new backup task policy has been planned based on the actual situation.

Procedure

Step 1 On MRS Manager, choose **System** > **Back Up Data**.

Step 2 In the task list, locate a specified task, click **Modify** in the **Operation** column to go to the configuration modification page.

Step 3 Modify the following parameters on the displayed page:

- Manual backup:
  - Target Path
  - Max Number of Backup Copies
- Periodic backup:
  - Started
  - Period
  - Target Path
  - Max Number of Backup Copies

  NOTE

  - When **Path Type** is set to **LocalHDFS**, **Target Path** is valid for modifying a backup task.
  - After you change the value of **Target Path** for a backup task, full backup is performed by default when the task is executed for the first time.

Step 4 Click **OK**.

----End

11.10.5 Viewing Backup and Restoration Tasks

Scenario

This section describes how to view created backup and restoration tasks and check their running status on MRS Manager.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 Click **Back Up Data** or **Restore Data**.

Step 3 In the task list, obtain the previous execution result in the **Task Progress** column. Green indicates that the task is executed successfully, and red indicates that the execution fails.

Step 4 In the **Operation** column of a specified task in the task list, choose **More** > **View History** to view the historical record of backup and restoration execution.

In the displayed window, click **View** in the **Details** column. The task execution logs and paths are displayed.

----End

Related Tasks

- Modifying a backup task

  For details, see **Modifying a Backup Task**.
- Viewing a restoration task

  In the **Operation** column of the specified task in the task list, click **View Details** to view the restoration task. You can only view but cannot modify the parameters of a restoration task.
- Executing a backup or restoration task

  In the task list, locate a specified task and click **Start** in the **Operation** column to start a backup or restoration task that is ready or fails to be executed. Executed restoration tasks cannot be repeatedly executed.
- Stopping backup tasks

  In the task list, locate a specified task and click **More** > **Stop** in the **Operation** column to stop a backup task that is running.
- Deleting a backup or restoration task

  In the **Operation** column of the specified task in the task list, choose **More** > **Delete** to delete the backup or restoration task. After a task is deleted, the backup data is retained by default.
- Suspending a backup task

  In the **Operation** column of the specified task in the task list, choose **More** > **Suspend** to suspend the backup task. Only periodic backup tasks can be suspended. Suspended backup tasks are no longer executed automatically. When you suspend a backup task that is being executed, the task execution stops. To cancel the suspension status of a task, click **More** > **Resume**.

11.11 Security Management

11.11.1 Default Users of Clusters with Kerberos Authentication Disabled

User Classification

The MRS cluster provides the following two types of users. Users are advised to periodically change the passwords. It is not recommended to use the default passwords.

User Type	Description
System users	User who runs OMS processes
Database users	• User who manages OMS database and accesses data • User who runs the database of service components (Hive, Loader, and DBService)

System users

NOTE

- User **Idap** of the OS is required in the MRS cluster. Do not delete this account. Otherwise, the cluster may not work properly. Password management policies are maintained by the operation users.
- Reset the passwords when you change the passwords of user **ommdba** and user **omm** for the first time. Change the passwords periodically after retrieving them.

Operation	Username	Initial Password	Description
System administrator of the MRS cluster	admin	Specified by the user during the cluster creation	MRS Manager administrator. This user also has the following permissions: • Common HDFS and ZooKeeper user permissions. • Permissions to submit and query MapReduce and Yarn tasks, manage Yarn queues, and access the Yarn web UI. • Permissions to submit, query, activate, deactivate, reassign, delete topologies, and operate all topologies of the Storm service. • Permissions to create, delete, authorize, reassign, consume, write, and query topics of the Kafka service.

Operation	Username	Initial Password	Description
MRS cluster node OS user	omm	Randomly generated by the system	Internal running user of the MRS cluster system. This user is an OS user generated on all node and does not require a unified password.
MRS cluster node OS user	root	Set by the user	User for logging in to the node in the MRS cluster. This user is an OS user generated on all nodes.

User Group Information

Default User Group	Description
supergroup	Primary group of user **admin**, which has no additional permissions in the cluster with Kerberos authentication disabled.
check_sec_ldap	Used to test whether the active LDAP works properly. This user group is generated randomly in a test and automatically deleted after the test is complete. which is an internal system user group used only between components.
Manager_tenant	Tenant system user group, which is an internal system user group used only between components. It is used only in clusters with Kerberos authentication enabled.
System_administrator	MRS cluster system administrator group, which is an internal system user group used only between components. It is used only in clusters with Kerberos authentication enabled.
Manager_viewer	MRS Manager system viewer group, which is an internal system user group used only between components. It is used only in clusters with Kerberos authentication enabled.
Manager_operator	MRS Manager system operator group, which is an internal system user group used only between components. It is used only in clusters with Kerberos authentication enabled.
Manager_auditor	MRS Manager system auditor group, which is an internal system user group used only between components. It is used only in clusters with Kerberos authentication enabled.

Default User Group	Description
Manager_administrator	MRS Manager system administrator group, which is an internal system user group used only between components. It is used only in clusters with Kerberos authentication enabled.
compcommon	MRS cluster internal group, used to access public resources in the cluster. All system users and system running users are added to this user group by default.
default_1000	User group created for tenants, which is an internal system user group used only between components.
launcher-job	MRS internal group, which is used to submit jobs using V2 APIs.

OS User Group	Description
wheel	Primary group of MRS internal running user **omm**.
ficommon	MRS cluster common group that corresponds to **compcommon** for accessing public resource files stored in the OS of the cluster.

Database users

MRS cluster system database users include OMS database users and DBService database users.

NOTE

Do not delete database users. Otherwise, the cluster or components may not work properly.

Operation	Default User	Initial Password	Description
OMS database	ommdba	dbChangeMe@123456	OMS database administrator who performs maintenance operations, such as creating, starting, and stopping.
	omm	ChangeMe@123456	User for accessing OMS database data
DBService database	omm	dbserverAdmin@123	Administrator of the GaussDB database in the DBService component

Operation	Default User	Initial Password	Description
	hive	HiveUser@	User for Hive to connect to the DBService database
	hue	HueUser@123	User for Hue to connect to the DBService database
	sqoop	SqoopUser@	User for Loader to connect to the DBService database.

11.11.2 Default Users of Clusters with Kerberos Authentication Enabled

User Classification

The MRS cluster provides the following three types of users. Users are advised to periodically change the passwords. It is not recommended to use the default passwords.

User Type	Description
System user	• User created on Manager for MRS cluster O&M and service scenarios. There are two types of users: – **Human-machine** user: used for Manager O&M scenarios and component client operation scenarios. – **Machine-machine** user: used for MRS cluster application development scenarios. • User who runs OMS processes.
Internal system user	Internal user who performs process communications, saves user group information, and associates user permissions.
Database user	• User who manages OMS database and accesses data. • User who runs the database of service components (Hive, Hue, Loader, and DBService)

System User

NOTE

- User **Idap** of the OS is required in the MRS cluster. Do not delete this account. Otherwise, the cluster may not work properly. Password management policies are maintained by the operation users.
- Reset the passwords when you change the passwords of user **ommdba** and user **omm** for the first time. Change the passwords periodically after retrieving them.

Type	Username	Initial Password	Description
System administrator of the MRS cluster	admin	Specified by the user during the cluster creation.	Manager administrator with the following permissions: • Common HDFS and ZooKeeper user permissions. • Permissions to submit and query MapReduce and Yarn tasks, manage Yarn queues, and access the Yarn web UI. • Permissions to submit, query, activate, deactivate, reassign, delete topologies, and operate all topologies of the Storm service. • Permissions to create, delete, authorize, reassign, consume, write, and query topics of the Kafka service.
MRS cluster node OS user	omm	Randomly generated by the system.	Internal running user of the MRS cluster system. This user is an OS user generated on all nodes and does not require a unified password.
MRS cluster node OS user	root	Set by the user.	User for logging in to the node in the MRS cluster. This user is an OS user generated on all nodes.

Internal System Users

NOTE

Do not delete the following internal system users. Otherwise, the cluster or components may not work properly.

Type	Default User	Initial Password	Description
Component running user	hdfs	Hdfs@123	This user is the HDFS system administrator and has the following permissions: 1. File system operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. ● Views and sets disk quotas for users. 2. HDFS management operation permissions: ● Views the web UI status. ● Views and sets the active and standby HDFS status. ● Enters and exits the HDFS in security mode. ● Checks the HDFS file system.

Type	Default User	Initial Password	Description
	hbase	Hbase@123	This user is the HBase system administrator and has the following permissions: ● Cluster management permission: **Enable** and **Disable** operations on tables to trigger MajorCompact and ACL operations. ● Grants and revokes permissions, and shuts down the cluster. ● Table management permission: Creates, modifies, and deletes tables. ● Data management permission: Reads and writes data in tables, column families, and columns. ● Accesses the HBase web UI.
	mapred	Mapred@123	This user is the MapReduce system administrator and has the following permissions: ● Submits, stops, and views the MapReduce tasks. ● Modifies the Yarn configuration parameters. ● Accesses the Yarn and MapReduce web UI.
	spark	Spark@123	This user is the Spark system administrator and has the following permissions: ● Accesses the Spark web UI. ● Submits Spark tasks.

User Group Information

Default User Group	Description
hadoop	Users added to this user group have the permission to submit tasks to all Yarn queues.
hbase	Common user group. Users added to this user group will not have any additional permission.
hive	Users added to this user group can use Hive.
spark	Common user group. Users added to this user group will not have any additional permission.
supergroup	Users added to this user group can have the administrator permission of HBase, HDFS, and Yarn and can use Hive.
check_sec_ldap	Used to test whether the active LDAP works properly. This user group is generated randomly in a test and automatically deleted after the test is complete. This is an internal system user group used only between components.
Manager_tenant	Tenant system user group, which is an internal system user group used only between components.
System_administrator	MRS cluster system administrator group, which is an internal system user group used only between components.
Manager_viewer	MRS Manager system viewer group, which is an internal system user group used only between components.
Manager_operator	MRS Manager system operator group, which is an internal system user group used only between components.
Manager_auditor	MRS Manager system auditor group, which is an internal system user group used only between components.
Manager_administrator	MRS Manager system administrator group, which is an internal system user group used only between components.
compcommon	Internal system group for accessing public resources in a cluster. All system users and system running users are added to this user group by default.
default_1000	User group created for tenants, which is an internal system user group used only between components.

Default User Group	Description
kafka	Kafka common user group. Users added to this group need to be granted with read and write permission by users in the **kafkaadmin** group before accessing the desired topics.
kafkasuperuser	Users added to this group have permissions to read data from and write data to all topics.
kafkaadmin	Kafka administrator group. Users added to this group have the permissions to create, delete, authorize, as well as read from and write data to all topics.
storm	Storm common user group. Users added to this group have the permissions to submit topologies and manage their own topologies.
stormadmin	Storm administrator user group. Users added to this group have the permissions to submit topologies and manage their own topologies.
opentsdb	Common user group. Users added to this user group will not have any additional permission.
presto	Common user group. Users added to this user group will not have any additional permission.
flume	Common user group. Users added to this user group will not have any additional permission.
launcher-job	MRS internal group, which is used to submit jobs using V2 APIs.

OS User Group	Description
wheel	Primary group of MRS internal running user **omm**.
ficommon	MRS cluster common group that corresponds to **compcommon** for accessing public resource files stored in the OS of the cluster.

Database User

MRS cluster system database users include OMS database users and DBService database users.

NOTE

Do not delete database users. Otherwise, the cluster or components may not work properly.

Type	Default User	Initial Password	Description
OMS database	ommdba	dbChangeMe@123456	OMS database administrator who performs maintenance operations, such as creating, starting, and stopping applications.
	omm	ChangeMe@123456	User for accessing OMS database data.
DBService database	omm	dbserverAdmin@123	Administrator of the GaussDB database in the DBService component.
	hive	HiveUser@	User for Hive to connect to the DBService database.
	hue	HueUser@123	User for Hue to connect to the DBService database.
	sqoop	SqoopUser@	User for Loader to connect to the DBService database.
	ranger	RangerUser@	User for Ranger to connect to the DBService database.

11.11.3 Changing the Password of an OS User

Scenario

This section describes how to periodically change the login passwords of the OS users **omm**, **ommdba**, and **root** on MRS cluster nodes to improve the system O&M security.

Passwords of users **omm**, **ommdba**, and **root** on each node can be different.

Procedure

Step 1 Log in to the **Master1** node and then log in to other nodes whose OS user passwords need to be changed.

Step 2 Run the following command to switch to user **root**:

sudo su - root

Step 3 Run the following command to change the passwords of users **omm**, **ommdba**, or **root**:

passwd omm

passwd ommdba

passwd root

For example, if you run the **omm:passwd** command, the system displays the following information:

```
Changing password for user omm.
New password:
```

Enter a new password. The password change policies for an OS vary according to the OS that is used.

```
Retype new password:
passwd: all authentication tokens updated successfully.
```

NOTE

The default password complexity requirements of the MRS cluster are as follows:

- The password must contain at least eight characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$%^&*()-_=+\|[{}];:'",<.>/?).
- The new password cannot be the same as last five historical passwords.

----End

11.11.4 Changing the password of user admin

This section describes how to periodically change the password of cluster user **admin** to improve the system O&M security.

If the password is changed, the downloaded user credential will be unavailable. Download the authentication credential again, and replace the old one.

Changing the Password of User admin on the Cluster Node

Step 1 Update the client of the active management node. For details, see **Updating a Client (Versions Earlier Than 3.x)**.

Step 2 Log in to the active management node.

Step 3 (Optional) To change the password as user **omm**, run the following command to switch the user:

sudo su - omm

Step 4 Run the following command to switch to the client directory, for example, **/opt/client**.

cd /opt/client

Step 5 Run the following command to configure environment variables:

source bigdata_env

Step 6 Run the following command to change the password of user **admin**: This operation takes effect in the whole cluster.

kpasswd admin

Enter the old password and then enter a new password twice.

For the cluster, the default password complexity requirements are as follows:

- The password must contain at least eight characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

Changing the Password of User admin on MRS Manager

You can change the password of user **admin** on MRS Manager only for clusters with Kerberos authentication enabled and clusters with Kerberos authentication disabled but the EIP function enabled.

Step 1 Log in to MRS Manager as user **admin**.

Step 2 Click the username in the upper right corner of the page and choose **Change Password**.

Step 3 On the **Change Password** page, set **Old Password**, **New Password**, and **Confirm Password**.

NOTE

The default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$%^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

Step 4 Click **OK**. Log in to MRS Manager with the new password.

----End

Resetting the Password for User admin

Step 1 Log in to the **Master1** node.

Step 2 (Optional) To change the password as user **omm**, run the following command to switch the user:

sudo su - omm

Step 3 Run the following command to switch to the client directory, for example, **/opt/client**:

cd /opt/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 Run the following command to log in to the console as user **kadmin/admin**:

kadmin -p kadmin/admin

NOTE

The default password of user **kadmin/admin** is **KAdmin@123**, which will expire upon your first login. Change the password as prompted and keep the new password secure.

Step 6 Run the following command to reset the password of a component running user. This operation takes effect for all servers.

cpw *Component running user name*

For example, to reset the password of user admin, run the **cpw admin** command.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.11.5 Changing the Password of the Kerberos Administrator

Scenario

This section describes how to periodically change the password of the Kerberos administrator **kadmin** of the MRS cluster to improve the system O&M security.

If the password is changed, the downloaded user credential will be unavailable. Download the authentication credential again, and replace the old one.

Prerequisites

A client has been prepared on the **Master1** node.

Procedure

Step 1 Log in to the **Master1** node.

Step 2 (Optional) To change the password as user **omm**, run the following command to switch the user:

sudo su - omm

Step 3 Run the following command to switch to the client directory, for example, **/opt/client**.

cd /opt/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 Run the following command to change the password of **kadmin/admin**. This operation takes effect for all servers.

kpasswd kadmin/admin

For the cluster, the default password complexity requirements are as follows:

- The password must contain at least eight characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.11.6 Changing the Passwords of the LDAP Administrator and the LDAP User

Scenario

This section describes how to periodically change the passwords of the LDAP administrator **rootdn:cn=root,dc=hadoop,dc=com** and the LDAP user **pg_search_dn:cn=pg_search_dn,ou=Users,dc=hadoop,dc=com** to improve the system O&M security.

Impact on the System

All services need to be restarted for the new password to take effect. The services are unavailable during the restart.

Procedure

Step 1 On MRS Manager, choose **Services** > **LdapServer** > **More**.

Step 2 Click **Change Password**.

Step 3 In the **Change Password** dialog box, select the user whose password needs to be modified in the **User Information** drop-down box.

Step 4 Enter the old password in the **Old Password** text box, and enter the new password in the **New Password** and **Confirm Password** text boxes.

The default password complexity requirements are as follows:

- The password contains 16 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, and special characters (`~!@#$%^&*()-_=+\|[{}];:",<.>/?).
- The password cannot be the username or the reverse username.
- The new password cannot be the same as the current password.

NOTE

The default password of the LDAP administrator **rootdn:cn=root,dc=hadoop,dc=com** is **LdapChangeMe@123**, and that of the LDAP user **pg_search_dn:cn=pg_search_dn,ou=Users,dc=hadoop,dc=com** is **pg_search_dn@123**. Periodically change the passwords and keep them secure.

Step 5 Select **I have read the information and understand the impact**, and click **OK** to confirm the modification and restart the service.

----End

11.11.7 Changing the Password of a Component Running User

Scenario

This section describes how to periodically change the password of the component running user of the MRS cluster to improve the system O&M security.

If the initial password is randomly generated by the system, reset the password.

If the password is changed, the downloaded user credential will be unavailable. Download the authentication credential again, and replace the old one.

Prerequisites

A client has been prepared on the **Master1** node.

Procedure

Step 1 Log in to the **Master1** node.

Step 2 (Optional) To change the password as user **omm**, run the following command to switch the user:

sudo su - omm

Step 3 Run the following command to switch to the client directory, for example, **/opt/client**:

cd /opt/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 Run the following command to log in to the console as user **kadmin/admin**:

kadmin -p kadmin/admin

NOTE

The default password of user **kadmin/admin** is **KAdmin@123**, which will expire upon your first login. Change the password as prompted and keep the new password secure.

Step 6 Run the following command to reset the password of a component running user. This operation takes effect for all servers.

cpw *Component running user name*

For example, to reset the password of user admin, run the **cpw admin** command.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.

- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.11.8 Changing the Password of the OMS Database Administrator

Scenario

This section describes how to periodically change the password of the OMS database administrator to improve the system O&M security.

Procedure

Step 1 Log in to the active management node.

NOTE

The password of user **ommdba** cannot be changed on the standby management node. Otherwise, the cluster may not work properly. Change the password on the active management node only.

Step 2 Run the following command to switch the user:

sudo su - omm

Step 3 Run the following command to switch the directory:

cd $OMS_RUN_PATH/tools

Step 4 Run the following command to change the password of user **ommdba**:

mod_db_passwd ommdba

Step 5 Enter the old password of user **ommdba** and enter a new password twice.

The password complexity requirements are as follows:

- The password contains 16 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, and special characters ('~!@#$%^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.
- The password cannot be the same as the last 20 historical passwords.

If the following information is displayed, the password is changed successfully.

```
Congratulations, update [ommdba] password successfully.
```

----End

11.11.9 Changing the Password of the Data Access User of the OMS Database

Scenario

This section describes how to periodically change the password of the data access user of the OMS database to improve the system O&M security.

Impact on the System

The OMS service needs to be restarted for the new password to take effect. The service is unavailable during the restart.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Change OMS Database Password**.

Step 3 Locate the row that contains user **omm**, and click **Change password** in the **Operation** column.

The password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, and special characters ('~!@#$%^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.
- The password cannot be the same as the last 20 historical passwords.

Step 4 Click **OK**. When **Operation successful** is displayed, click **Finish**.

Step 5 Locate the row that contains user **omm**, and click **Restart the OMS service** in the **Operation** column to restart the OMS database.

NOTE

If the password is changed but the OMS database is not restarted, the status of user **omm** changes to **Waiting to restart** and the password cannot be changed until the OMS database is restarted.

Step 6 In the displayed dialog box, select **I have read the information and understand the impact**. Click **OK**, and restart the OMS service.

----End

11.11.10 Changing the Password of a Component Database User

Scenario

This section describes how to periodically change the password of the component database user to improve the system O&M security.

Impact on the System

The services need to be restarted for the new password to take effect. The services are unavailable during the restart.

Procedure

Step 1 On MRS Manager, click **Services** and click the name of the database user service to be modified.

Step 2 Determine the component database user whose password is to be changed.

- To change the password of the DBService database user, go to **Step 3**.
- To change the password of the Loader, Hive, or Hue database user, stop the service first and then execute **Step 3**.

Click **Stop Service**.

Step 3 Choose **More** > **Change Password**.

Step 4 Enter the old and new passwords as prompted.

The password complexity requirements are as follows:

- The password of the DBService database user contains 16 to 32 characters. The password of the Loader, Hive, or Hue database user contains 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, and special characters ('~!@#$%^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.
- The password cannot be the same as the last 20 historical passwords.

Step 5 Click **OK**. The system automatically restarts the corresponding service. When **Operation successful** is displayed, click **Finish**.

----End

11.11.11 Updating Cluster Keys

Scenario

When a cluster is installed, an encryption key is generated automatically to store the security information in the cluster (such as all database user passwords and key file access passwords) in encryption mode. After the cluster is successfully installed, you are advised to periodically update the encryption key based on the following procedure.

Impact on the System

- After a cluster key is updated, a new key is generated randomly in the cluster. This key is used to encrypt and decrypt the newly stored data. The old key is not deleted, and it is used to decrypt data encrypted using the old key. After security information is modified, for example, a database user password is changed, the new password is encrypted using the new key.

- When the key is updated, the cluster is stopped and cannot be accessed.

Prerequisites

The upper-layer applications depending on the cluster are stopped.

Procedure

Step 1 Log in to MRS Manager and choose **Services** > **More** > **Stop Cluster**.

In the displayed dialog box, select **I have read the information and understand the impact.** Click **OK**. Wait until the system displays a message indicating that the operation is successful. Click **Finish**. The cluster is stopped successfully.

Step 2 Log in to the active management node.

Step 3 Run the following commands to switch the user:

sudo su - omm

Step 4 Run the following command to disable logout upon timeout:

TMOUT=0

Step 5 Run the following command to switch the directory:

cd ${BIGDATA_HOME}/om-0.0.1/tools

Step 6 Run the following command to update the cluster key:

sh updateRootKey.sh

Enter **y** as prompted.

```
The root key update is a critical operation.
Do you want to continue?(y/n):
```

The key is updated successfully if the following information is displayed:

```
...
Step 4-1: The key save path is obtained successfully.
...
Step 4-4: The root key is sent successfully.
```

Step 7 On MRS Manager, choose **Services > More > Start Cluster**.

In the displayed dialog box, click **OK**. After **Operation successful** is displayed, click **Finish**. The cluster is started.

----End

11.12 Permissions Management

11.12.1 Creating a Role

Scenario

This section describes how to create a role on MRS Manager and authorize and manage Manager and components.

Up to 1,000 roles can be created on MRS Manager.

Prerequisites

You have learned service requirements.

Procedure

Step 1 On MRS Manager, choose **System** > **Manage Role**.

Step 2 Click **Create Role** and fill in **Role Name** and **Description**.

Role Name is mandatory and contains 3 to 30 digits, letters, and underscores (_). **Description** is optional.

Step 3 In **Permission**, set role permission.

1. Click **Service Name** and select a name in **View Name**.
2. Select one or more permissions.

NOTE

- The **Permission** parameter is optional.
- If you select **View Name** to set component permissions, you can enter a resource name in the **Search** box in the upper right corner and click . The search result is displayed.
- The search scope covers only directories with current permissions. You cannot search subdirectories. Search by keywords supports fuzzy match and is case-insensitive. Results of the next page can be searched.

Table 11-36 Manager permission description

Resource Supporting Permission Management	Permission Setting
Alarm	Authorizes the Manager alarm function. You can select **View** to view alarms and **Management** to manage alarms.
Audit	Authorizes the Manager audit log function. You can select **View** to view audit logs and **Management** to manage audit logs.
Dashboard	Authorizes the Manager overview function. You can select **View** to view the cluster overview.
Hosts	Authorizes the node management function. You can select **View** to view node information and **Management** to manage nodes.
Services	Authorizes the service management function. You can select **View** to view service information and **Management** to manage services.
System_cluster_management	Authorizes the MRS cluster management function. You can select **Management** to use the MRS patch management function.

Resource Supporting Permission Management	Permission Setting
System_configuration	Authorizes the MRS cluster configuration function. You can select **Management** to configure MRS clusters on Manager.
System_task	Authorizes the MRS cluster task function. You can select **Management** to manage periodic tasks of MRS clusters on Manager.
Tenant	Authorizes the Manager multi-tenant management function. You can select **Management** to manage multi-tenants.

Table 11-37 HBase permission description

Resource Supporting Permission Management	Permission Setting
SUPER_USER_GROUP	Grants you HBase administrator rights.
Global	HBase resource type, indicating the whole HBase.
Namespace	HBase resource type, indicating namespace, which is used to store HBase tables. It has the following permissions: • **Admin** permission to manage the namespace • **Create**: permission to create HBase tables in the namespace • **Read**: permission to access the namespace • **Write**: permission to write data to the namespace • **Execute**: permission to execute the coprocessor (Endpoint)
Table	HBase resource type, indicating a data table, which is used to store data. It has the following permissions: • **Admin**: permission to manage a data table • **Create**: permission to create column families and columns in a data table • **Read**: permission to read a data table • **Write**: permission to write data to a data table • **Execute**: permission to execute the coprocessor (Endpoint)

Resource Supporting Permission Management	Permission Setting
ColumnFamily	HBase resource type, indicating a column family, which is used to store data. It has the following permissions: ● **Create**: permission to create columns in a column family ● **Read**: permission to read a column family ● **Write**: permission to write data to a column family
Qualifier	HBase resource type, indicating a column, which is used to store data. It has the following permissions: ● **Read**: permission to read a column ● **Write**: permission to write data to a column

By default, permissions of an HBase resource type of each level are shared by resource types of sub-levels. However, the **Recursive** option is not selected by default. For example, if **Read** and **Write** permissions are added to the **default** namespace, they are automatically added to the tables, column families, and columns in the namespace. If a child resource is set after the parent resource, the permission of the child resource is the union of the permissions of the parent resource and the current child resource.

Table 11-38 HDFS permission description

Resource Supporting Permission Management	Permission Setting
Folder	HDFS resource type, indicating an HDFS directory, which is used to store files or subdirectories. It has the following permissions: ● **Read**: permission to access the HDFS directory ● **Write**: permission to write data to the HDFS directory ● **Execute**: permission to perform an operation. It must be selected when you add access or write permission.
Files	HDFS resource type, indicating a file in HDFS. It has the following permissions: ● **Read**: permission to access the file ● **Write**: permission to write data to the file ● **Execute**: permission to perform an operation. It must be selected when you add access or write permission.

Permissions of an HDFS directory of each level are not shared by directory types of sub-levels by default. For example, if **Read** and **Execute** permissions are added to the **tmp** directory, you must select **Recursive** at the same time to add permissions to subdirectories.

Table 11-39 Hive permission description

Resource Supporting Permission Management	Permission Setting
Hive Admin Privilege	Grants you Hive administrator rights.
Database	Hive resource type, indicating a Hive database, which is used to store Hive tables. It has the following permissions: • **Select**: permission to query the Hive database • **Delete**: permission to perform the deletion operation in the Hive database • **Insert**: permission to perform the insertion operation in the Hive database • **Create**: permission to perform the creation operation in the Hive database
Table	Hive resource type, indicating a Hive table, which is used to store data. It has the following permissions: • **Select**: permission to query the Hive table • **Delete**: permission to perform the deletion operation in the Hive table • **Update**: grants users the **Update** permission of the Hive table • **Insert**: permission to perform the insertion operation in the Hive table • **Grant of Select**: permission to grant the **Select** permission to other users using Hive statements • **Grant of Delete**: permission to grant the **Delete** permission to other users using Hive statements • **Grant of Update**: permission to grant the **Update** permission to other users using Hive statements • **Grant of Insert**: permission to grant the **Insert** permission to other users using Hive statements

By default, permissions of a Hive resource type of each level are shared by resource types of sub-levels. However, the **Recursive** option is not selected by default. For example, if **Select** and **Insert** permissions are added to the **default** database, they are automatically added to the tables and columns in the database. If a child resource is set after the parent resource, the permission of the child resource is the union of the permissions of the parent resource and the current child resource.

Table 11-40 Yarn permission description

Resource Supporting Permission Management	Permission Setting
Cluster Admin Operations	Grants you Yarn administrator rights.
root	Root queue of Yarn. It has the following permissions: • **Submit**: permission to submit jobs in the queue • **Admin**: permission to manage permissions of the current queue
Parent Queue	Yarn resource type, indicating a parent queue containing sub-queues. A root queue is a type of a parent queue. It has the following permissions: • **Submit**: permission to submit jobs in the queue • **Admin**: permission to manage permissions of the current queue
Leaf Queue	Yarn resource type, indicating a leaf queue. It has the following permissions: • **Submit**: permission to submit jobs in the queue • **Admin**: permission to manage permissions of the current queue

By default, permissions of a Yarn resource type of each level are shared by resource types of sub-levels. However, the **Recursive** option is not selected by default. For example, if the **Submit** permission is added to the **root** queue, it is automatically added to the sub-queue. Permissions inherited by sub-queues will not be displayed as selected in the **Permission** table. If a child resource is set after the parent resource, the permission of the child resource is the union of the permissions of the parent resource and the current child resource.

Table 11-41 Hue permission description

Resource Supporting Permission Management	Permission Setting
Storage Policy Admin	Grants you storage policy administrator rights.

Step 4 Click **OK**. Return to **Manage Role**.

----End

Related Tasks

Modifying a role

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage Role**.

Step 3 In the row of the role to be modified, click **Modify** to modify role information.

NOTE

If you change permissions assigned by the role, it takes 3 minutes to make new configurations take effect.

Step 4 Click **OK**. The modification is complete.

----End

Deleting a role

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage Role**.

Step 3 In the row of the role to be deleted, click **Delete**.

Step 4 Click **OK**. The role is deleted.

----End

11.12.2 Creating a User Group

Scenario

This section describes how to create user groups and specify their operation permissions on MRS Manager. Management of single or multiple users can be unified in the user groups. After being added to a user group, users can obtain operation permissions owned by the user group.

Up to 100 user groups can be created on MRS Manager.

Prerequisites

Administrators have learned service requirements and created roles required by service scenarios.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User Group**.

Step 3 Above the user group list, click **Create User Group**.

Step 4 Input **Group Name** and **Description**.

Group Name is mandatory and contains 3 to 20 digits, letters, and underscores (_). **Description** is optional.

Step 5 In **Role**, click **Select and Add Role** to select and add specified roles.

If you do not add the roles, the user group you are creating now does not have the permission to use MRS clusters.

Step 6 Click **OK**. The user group is created.

----End

Related Tasks

Modifying a user group

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User Group**.

Step 3 In the row of the user group to be modified, click **Modify**.

NOTE

If you change role permissions assigned to the user group, it takes 3 minutes to make new configurations take effect.

Step 4 Click **OK**. The modification is complete.

----End

Deleting a user group

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User Group**.

Step 3 In the row of the user group to be deleted, click **Delete**.

Step 4 Click **OK**. The user group is deleted.

----End

11.12.3 Creating a User

Scenario

This section describes how to create users on MRS Manager based on site requirements and specify their operation permissions to meet service requirements.

Up to 1,000 users can be created on MRS Manager.

If a new password policy needs to be used for a new user's password, follow instructions in **Modifying a Password Policy** to modify the password policy and then perform the following operations to create a user.

Prerequisites

Administrators have learned service requirements and created roles and role groups required by service scenarios.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 Above the user list, click **Create User**.

Step 4 Configure parameters as prompted and enter a username in **User Name**.

NOTE

- If a username exists, you cannot create another username that only differs from the existing username in case. For example, if **User1** has been created, you cannot create **user1**.
- When you use the user you created, enter the correct username, which is case-sensitive.
- **User Name** is mandatory and contains 3 to 20 digits, letters, and underscores (_).
- **root**, **omm**, and **ommdba** are reserved system user. Select another username.

Step 5 Set **User Type** to either **Human-Machine** or **Machine-Machine**.

- **Human-Machine** users: used for O&M on MRS Manager and operations on component clients. If you select this user type, you need to enter a password and confirm the password in **Password** and **Confirm Password** accordingly.
- **Machine-Machine** users: used for MRS application development. If you select this user type, you do not need to enter a password, because the password is randomly generated.

Step 6 In **User Group**, click **Select and Join User Group** to select user groups and add users to them.

NOTE

- If roles have been added to user groups, the users can be granted with permissions of the roles.
- If you want to grant new users with Hive permissions, add the users to the Hive group.
- If a user needs to manage tenant resources, the user group must be assigned the **Manager_tenant** role and the role corresponding to the tenant.

Step 7 In **Primary Group**, select a group as the primary group for users to create directories and files. The drop-down list contains all groups selected in **User Group**.

Step 8 In **Assign Rights by Role**, click **Select and Add Role** to add roles for users based on service requirements.

NOTE

- When you create a user, if permissions of a user group that is granted to the user cannot meet service requirements, you can assign other created roles to the user. It takes 3 minutes to make role permissions granted to the new user take effect.
- Adding a role when you create a user can specify the user rights.
- A new user can access WebUIs of HDFS, HBase, Yarn, Spark, and Hue even when roles are not assigned to the user.

Step 9 In **Description**, provide description based on onsite service requirements.

Description is optional.

Step 10 Click **OK**. The user is created.

If a new user is used in the MRS cluster for the first time, for example, used for logging in to MRS Manager or using the cluster client, the password must be changed. For details, see section **Changing the Password of an Operation User**.

----End

11.12.4 Modifying User Information

Scenario

This section describes how to modify user information on MRS Manager, including information about the user group, primary group, role, and description.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 In the row of the user to be modified, click **Modify**.

NOTE

If you change user groups for or assign role permissions to the user, it takes 3 minutes to make new configurations take effect.

Step 4 Click **OK**. The modification is complete.

----End

11.12.5 Locking a User

This section describes how to lock users in MRS clusters. A locked user cannot log in to MRS Manager or perform security authentication in the cluster.

A locked user can be unlocked by an administrator manually or until the lock duration expires. You can lock a user by using either of the following methods:

- Automatic lock: Set **Number of Password Retries** in **Configure Password Policy**. If user login attempts exceed the parameter value, the user is automatically locked. For details, see **Modifying a Password Policy** .
- Manual lock: The administrator manually locks a user.

The following describes how to manually lock a user. **Machine-Machine** users cannot be locked.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 In the row of the user to be locked, click **Lock User**.

Step 4 In the window that is displayed, click **Yes** to lock the user.

----End

11.12.6 Unlocking a User

If a user is locked because the number of login attempts exceeds the value of **Number of Password Retries**, or the user is manually locked by the administrator, the administrator can unlock the user on MRS Manager.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 In the row of the user to be unlocked, click **Unlock User**.

Step 4 In the window that is displayed, click **Yes** to unlock the user.

----End

11.12.7 Deleting a User

Scenario

If an MRS cluster user is not required, the administrator can delete the user on MRS Manager.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 In the row of the user to be deleted, choose **More** > **Delete**.

Step 4 Click **OK**.

----End

11.12.8 Changing the Password of an Operation User

Scenario

Passwords of **Human-Machine** system users must be regularly changed to ensure MRS cluster security. This section describes how to change your passwords on MRS Manager.

If a new password policy needs to be used for the password modified by the user, follow instructions in **Modifying a Password Policy** to modify the password policy and then perform the following operations to modify the password.

Impact on the System

If you have downloaded a user authentication file, download it again and obtain the keytab file after changing the password of the MRS cluster user.

Prerequisites

- You have obtained the current password policies from the administrator.
- You have obtained the MRS Manager access address from the administrator.

Procedure

Step 1 On MRS Manager, move the mouse cursor to  in the upper right corner.

On the menu that is displayed, select **Change Password**.

Step 2 Fill in the **Old Password**, **New Password**, and **Confirm Password**. Click **OK**.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.12.9 Initializing the Password of a System User

Scenario

This section describes how to initialize a password on MRS Manager if a user forgets the password or the password of a public account needs to be changed regularly. After password initialization, the user must change the password upon the first login.

Impact on the System

If you have downloaded a user authentication file, download it again and obtain the keytab file after initializing the password of the MRS cluster user.

Initializing the Password of a Human-Machine User

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 Locate the row that contains the user whose password is to be initialized, choose **More** > **Initialize password**, and change the password as prompted.

In the window that is displayed, enter the password of the current administrator account and click **OK**. Then in **Initialize password**, click **OK**.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).

- The password cannot be the username or the reverse username.

----End

Initializing the Password of a Machine-Machine User

Step 1 Prepare a client based on service conditions and log in to the node where the client is installed.

Step 2 Run the following command to switch the user:

sudo su - omm

Step 3 Run the following command to switch to the client directory, for example, **/opt/client**:

cd /opt/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 Run the following command to log in to the console as user **kadmin/admin**:

kadmin -p kadmin/admin

NOTE

The default password of user **kadmin/admin** is **KAdmin@123**, which will expire upon your first login. Change the password as prompted and keep the new password secure.

Step 6 Run the following command to reset the password of a component running user. This operation takes effect for all servers.

cpw *Component running user name*

For example, **cpw oms/manager**.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.12.10 Downloading a User Authentication File

Scenario

When a user develops big data applications and runs them in an MRS cluster that supports Kerberos authentication, the user needs to prepare a user authentication file for accessing the MRS cluster. The keytab file in the authentication file can be used for user authentication.

This section describes how to download a user authentication file and export the keytab file on MRS Manager.

NOTE

- Before downloading a **Human-machine** user authentication file, change the password for the user on MRS Manager to make the initial password set by the administrator invalid. Otherwise, the exported keytab file cannot be used. For details, see **Changing the Password of an Operation User**.
- After a user password is changed, the exported keytab file becomes invalid, and you need to export a keytab file again.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User**.

Step 3 In the row of the user for whom you want to export the keytab file, choose **More** > **Download authentication credential** to download the authentication file. After the file is automatically generated, save it to a specified path and keep it properly.

Step 4 Open the authentication file with a decompression program.

- **user.keytab** indicates a user keytab file used for user authentication.
- **krb5.conf** indicates the configuration file of the authentication server. The application connects to the authentication server according to the configuration file information when authenticating users.

----End

11.12.11 Modifying a Password Policy

Scenario

This section describes how to set password and user login security rules as well as user lock rules. Password policies set on MRS Manager take effect for **Human-machine** users only, because the passwords of **Machine-machine** users are randomly generated.

If a new password policy needs to be used for a new user's password or the password modified by the user, perform the following operations to modify the password policy first, and then create a user or change the password by following instructions in **Creating a User** or **Changing the Password of an Operation User**.

NOTICE

Modify password policies based on service security requirements, because they involve user management security. Otherwise, security risks may be caused.

Procedure

Step 1 On MRS Manager, click **System**.

Step 2 Click **Configure Password Policy**.

Step 3 Modify password policies as prompted. For parameter details, see the following table:

Table 11-42 Password policy parameter description

Parameter	Description
Minimum Password Length	Indicates the minimum number of characters a password contains. The value ranges from 8 to 32. The default value is **8**.
Number of Character Types	Indicates the minimum number of character types a password contains. The character types are uppercase letters, lowercase letters, digits, spaces, and special characters (~`!?,.:;-_'(){}[]/<>@#$ %^&*+|\\=). The value can be **3** or **4**. The default value **3** indicates that the password must contain at least three types of the following characters: uppercase letters, lowercase letters, digits, special characters, and spaces.
Password Validity Period (days)	Indicates the validity period (days) of a password. The value ranges from 0 to 90. 0 means that the password is permanently valid. The default value is **90**.
Password Expiration Notification Days	Indicates the number of days in advance users are notified that their passwords are about to expire. After the value is set, if the difference between the cluster time and the password expiration time is smaller than this value, the user receives password expiration notifications. When a user logs in to MRS Manager, a message is displayed, indicating that the password is about to expire and asking the user whether to change the password. The value ranges from **0** to *X* (*X* must be set to the half of the password validity period and rounded down). Value **0** indicates that no notification is sent. The default value is **5**.
Interval of Resetting Authentication Failure Count (min)	Indicates the interval of retaining incorrect password attempts, in minutes. The value ranges from 0 to 1440. 0 indicates that incorrect password attempts are permanently retained and 1440 indicates that incorrect password attempts are retained for one day. The default value is **5**.
Number of Password Retries	Indicates the number of consecutive wrong passwords allowed before the system locks the user. The value ranges from 3 to 30. The default value is **5**.

Parameter	Description
Account Lock Duration (min)	Indicates the time period for which a user is locked when the user lockout conditions are met. The value ranges from 5 to 120. The default value is **5**.

----End

11.13 MRS Multi-User Permission Management

11.13.1 Users and Permissions of MRS Clusters

Overview

- **MRS Cluster Users**

  Indicate the security accounts of Manager, including usernames and passwords. These accounts are used to access resources in MRS clusters. Each MRS cluster in which Kerberos authentication is enabled can have multiple users.

- **MRS Cluster Roles**

  Before using resources in an MRS cluster, users must obtain the access permission which is defined by MRS cluster objects. A cluster role is a set of one or more permissions. For example, the permission to access a directory in HDFS needs to be configured in the specified directory and saved in a role.

Manager provides the user permission management function for MRS clusters, facilitating permission and user management.

- Permission management: adopts the role-based access control (RBAC) mode. In this mode, permissions are granted by role to form a permission set. After one or more roles are allocated to a user, the user can obtain the permissions of the roles.
- User management: uses MRS Manager to uniformly manage users, adopts the Kerberos protocol for user identity verification, and employs Lightweight Directory Access Protocol (LDAP) to store user information.

Permission Management

Permissions provided by MRS clusters include the O&M permissions of Manager and components (such as HDFS, HBase, Hive, and Yarn). In actual application, permissions must be assigned to each user based on service scenarios. To facilitate permission management, Manager introduces the role function to allow administrators to select and assign specified permissions. Permissions are centrally viewed and managed in permission sets, enhancing user experience.

A role is a logical entity that contains one or more permissions. Permissions are assigned to roles, and users can be granted the permissions by obtaining the roles.

A role can have multiple permissions, and a user can be bound to multiple roles.

- Role 1: is assigned operation permissions A and B. After role 1 is allocated to users a and b, users a and b can obtain operation permissions A and B.
- Role 2: is assigned operation permission C. After role 2 is allocated to users c and d, users c and d can obtain operation permission C.
- Role 3: is assigned operation permissions D and F. After role 3 is allocated to user a, user a can obtain operation permissions D and F.

For example, if an MRS user is bound to the administrator role, the user becomes an administrator of the MRS cluster.

Table 11-43 lists the roles that are created by default on Manager.

Table 11-43 Default roles and description

Default Role	Description
default	Tenant role
Manager_administrator	Manager administrator: This role has the permission to manage MRS Manager.
Manager_auditor	Manager auditor: This role has the permission to view and manage auditing information.
Manager_operator	Manager operator: This role has all permissions except tenant, configuration, and cluster management permissions.
Manager_viewer	Manager viewer: This role has the permission to view the information about systems, services, hosts, alarms, and auditing logs.
System_administrator	System administrator: This role has the permissions of Manager administrators and all service administrators.
Manager_tenant	Manager tenant viewer: This role has the permission to view information on the **Tenant** page on MRS Manager.

When creating a role on Manager, you can perform rights management for Manager and components, as shown in **Table 11-44**.

Table 11-44 Manager and component permission management

Permission	Description
Manager	Manager access and login permission.
HBase	HBase administrator permission and permission for accessing HBase tables and column families.
HDFS	HDFS directory and file permission.

Permission	Description
Hive	• Hive Admin Privilege Hive administrator permission. • Hive Read Write Privileges Hive data table management permission to set and manage the data of created tables.
Hue	Storage policy administrator permissions.
Yarn	• Cluster Admin Operations Yarn administrator permission. • Scheduler Queue Queue resource management permission.

User Management

MRS clusters that support Kerberos authentication use the Kerberos protocol and LDAP for user management.

- Kerberos verifies the identity of the user when a user logs in to Manager or uses a component client. Identity verification is not required for clusters with Kerberos authentication disabled.
- LDAP is used to store user information, including user records, user group information, and permission information.

MRS clusters can automatically update Kerberos and LDAP user data when users are created or modified on Manager. They can also automatically perform user identity verification and authentication and obtain user information when a user logs in to Manager or uses a component client. This ensures the security of user management and simplifies the user management tasks. Manager also provides the user group function for managing one or multiple users by type:

- A user group is a set of users, which can be used to manage users by type. Users in the system can exist independently or in a user group.
- After a user is added to a user group to which roles are allocated, the role permission of the user group is assigned to the user.

Table 11-45 lists the user groups that are created by default on MRS Manager in MRS 3.x or earlier.

For details about the default user groups displayed on FusionInsight Manager of MRS 3.*x* or later, see **User group**.

Table 11-45 Default user groups and description

User Group	Description
hadoop	Users added to this user group have the permission to submit tasks to all Yarn queues.

User Group	Description
hbase	Common user group. Users added to this user group will not have any additional permission.
hive	Users added to this user group can use Hive.
spark	Common user group. Users added to this user group will not have any additional permission.
supergroup	Users added to this user group can have the administrator permission of HBase, HDFS, and Yarn and can use Hive.
flume	Common user group. Users added to this user group will not have any additional permission.
kafka	Kafka common user group. Users added to this group need to be granted with read and write permission by users in the **kafkaadmin** group before accessing the desired topics.
kafkasuperuser	Users added to this group have permissions to read data from and write data to all topics.
kafkaadmin	Kafka administrator group. Users added to this group have the permissions to create, delete, authorize, as well as read from and write data to all topics.
storm	Storm common user group. Users added to this group have the permissions to submit topologies and manage their own topologies.
stormadmin	Storm administrator user group. Users added to this group have the permissions to submit topologies and manage their own topologies.

User **admin** is created by default for MRS clusters with Kerberos authentication enabled and is used for administrators to maintain the clusters.

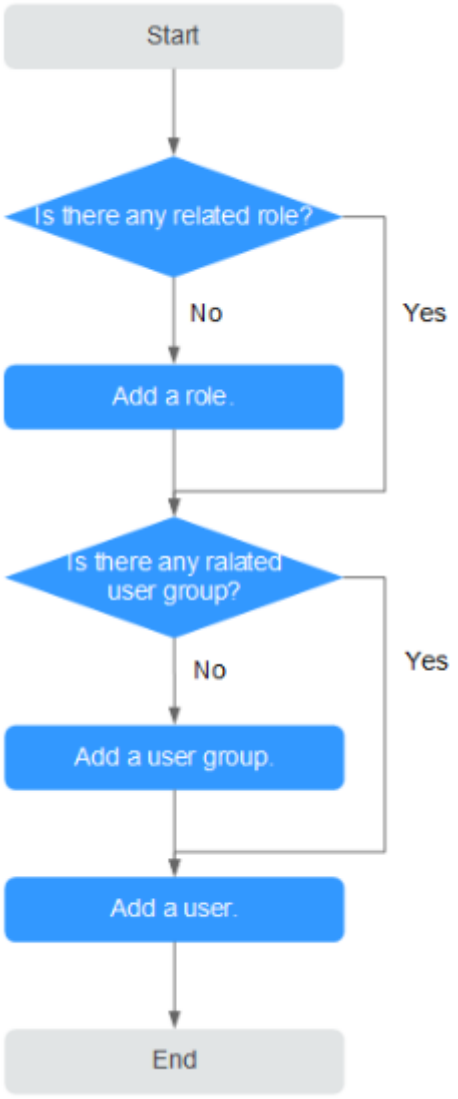
Process Overview

In practice, MRS cluster users must understand the service scenarios of big data and plan user permissions. Then, create roles and assign permissions to the roles on MRS Manager to meet service requirements. Manager provides the user group function for administrators to create user groups for managing users of one or multiple service scenarios of the same type.

NOTE

If a role has the permission of HDFS, HBase, Hive, or Yarn respectively, the role can only use the corresponding functions of the component. To use Manager, the corresponding Manager permission must be added to the role.

Figure 11-1 Process of creating a user



11.13.2 Default Users of Clusters with Kerberos Authentication Enabled

User Classification

The MRS cluster provides the following three types of users. Users are advised to periodically change the passwords. It is not recommended to use the default passwords.

User Type	Description
System user	• User created on Manager for MRS cluster O&M and service scenarios. There are two types of users: – **Human-machine** user: used for Manager O&M scenarios and component client operation scenarios. – **Machine-machine** user: used for MRS cluster application development scenarios. • User who runs OMS processes.
Internal system user	Internal user who performs process communications, saves user group information, and associates user permissions.
Database user	• User who manages OMS database and accesses data. • User who runs the database of service components (Hive, Hue, Loader, and DBService)

System User

NOTE

- User **Idap** of the OS is required in the MRS cluster. Do not delete this account. Otherwise, the cluster may not work properly. Password management policies are maintained by the operation users.
- Reset the passwords when you change the passwords of user **ommdba** and user **omm** for the first time. Change the passwords periodically after retrieving them.

Type	Username	Initial Password	Description
System administrator of the MRS cluster	admin	Specified by the user during the cluster creation.	Manager administrator with the following permissions: • Common HDFS and ZooKeeper user permissions. • Permissions to submit and query MapReduce and Yarn tasks, manage Yarn queues, and access the Yarn web UI. • Permissions to submit, query, activate, deactivate, reassign, delete topologies, and operate all topologies of the Storm service. • Permissions to create, delete, authorize, reassign, consume, write, and query topics of the Kafka service.
MRS cluster node OS user	omm	Randomly generated by the system.	Internal running user of the MRS cluster system. This user is an OS user generated on all nodes and does not require a unified password.
MRS cluster node OS user	root	Set by the user.	User for logging in to the node in the MRS cluster. This user is an OS user generated on all nodes.

Internal System Users

NOTE

Do not delete the following internal system users. Otherwise, the cluster or components may not work properly.

Type	Default User	Initial Password	Description
Component running user	hdfs	Hdfs@123	This user is the HDFS system administrator and has the following permissions: 1. File system operation permissions: ● Views, modifies, and creates files. ● Views and creates directories. ● Views and modifies the groups where files belong. ● Views and sets disk quotas for users. 2. HDFS management operation permissions: ● Views the web UI status. ● Views and sets the active and standby HDFS status. ● Enters and exits the HDFS in security mode. ● Checks the HDFS file system.

Type	Default User	Initial Password	Description
	hbase	Hbase@123	This user is the HBase system administrator and has the following permissions: ● Cluster management permission: **Enable** and **Disable** operations on tables to trigger MajorCompact and ACL operations. ● Grants and revokes permissions, and shuts down the cluster. ● Table management permission: Creates, modifies, and deletes tables. ● Data management permission: Reads and writes data in tables, column families, and columns. ● Accesses the HBase web UI.
	mapred	Mapred@123	This user is the MapReduce system administrator and has the following permissions: ● Submits, stops, and views the MapReduce tasks. ● Modifies the Yarn configuration parameters. ● Accesses the Yarn and MapReduce web UI.
	spark	Spark@123	This user is the Spark system administrator and has the following permissions: ● Accesses the Spark web UI. ● Submits Spark tasks.

User Group Information

Default User Group	Description
hadoop	Users added to this user group have the permission to submit tasks to all Yarn queues.
hbase	Common user group. Users added to this user group will not have any additional permission.
hive	Users added to this user group can use Hive.
spark	Common user group. Users added to this user group will not have any additional permission.
supergroup	Users added to this user group can have the administrator permission of HBase, HDFS, and Yarn and can use Hive.
check_sec_ldap	Used to test whether the active LDAP works properly. This user group is generated randomly in a test and automatically deleted after the test is complete. This is an internal system user group used only between components.
Manager_tenant	Tenant system user group, which is an internal system user group used only between components.
System_administrator	MRS cluster system administrator group, which is an internal system user group used only between components.
Manager_viewer	MRS Manager system viewer group, which is an internal system user group used only between components.
Manager_operator	MRS Manager system operator group, which is an internal system user group used only between components.
Manager_auditor	MRS Manager system auditor group, which is an internal system user group used only between components.
Manager_administrator	MRS Manager system administrator group, which is an internal system user group used only between components.
compcommon	Internal system group for accessing public resources in a cluster. All system users and system running users are added to this user group by default.
default_1000	User group created for tenants, which is an internal system user group used only between components.

Default User Group	Description
kafka	Kafka common user group. Users added to this group need to be granted with read and write permission by users in the **kafkaadmin** group before accessing the desired topics.
kafkasuperuser	Users added to this group have permissions to read data from and write data to all topics.
kafkaadmin	Kafka administrator group. Users added to this group have the permissions to create, delete, authorize, as well as read from and write data to all topics.
storm	Storm common user group. Users added to this group have the permissions to submit topologies and manage their own topologies.
stormadmin	Storm administrator user group. Users added to this group have the permissions to submit topologies and manage their own topologies.
opentsdb	Common user group. Users added to this user group will not have any additional permission.
presto	Common user group. Users added to this user group will not have any additional permission.
flume	Common user group. Users added to this user group will not have any additional permission.
launcher-job	MRS internal group, which is used to submit jobs using V2 APIs.

OS User Group	Description
wheel	Primary group of MRS internal running user **omm**.
ficommon	MRS cluster common group that corresponds to **compcommon** for accessing public resource files stored in the OS of the cluster.

Database User

MRS cluster system database users include OMS database users and DBService database users.

NOTE

Do not delete database users. Otherwise, the cluster or components may not work properly.

Type	Default User	Initial Password	Description
OMS database	ommdba	dbChangeMe@123456	OMS database administrator who performs maintenance operations, such as creating, starting, and stopping applications.
	omm	ChangeMe@123456	User for accessing OMS database data.
DBService database	omm	dbserverAdmin@123	Administrator of the GaussDB database in the DBService component.
	hive	HiveUser@	User for Hive to connect to the DBService database.
	hue	HueUser@123	User for Hue to connect to the DBService database.
	sqoop	SqoopUser@	User for Loader to connect to the DBService database.
	ranger	RangerUser@	User for Ranger to connect to the DBService database.

11.13.3 Creating a Role

Scenario

This section describes how to create a role on Manager and authorize and manage Manager and components.

Up to 1000 roles can be created on Manager.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Managing Roles**.

Prerequisites

- You have learned service requirements.
- You have obtained a cluster with Kerberos authentication enabled or a common cluster with the EIP function enabled.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, choose **System** > **Manage Role**.

Step 3 Click **Create Role** and fill in **Role Name** and **Description**.

Role Name is mandatory and contains 3 to 30 characters. Only digits, letters, and underscores (_) are allowed. **Description** is optional.

Step 4 In **Permission**, set role permission.

1. Click **Service Name** and select a name in **View Name**.
2. Select one or more permissions.

NOTE

- The **Permission** parameter is optional.
- If you select **View Name** to set component permissions, you can enter a resource name in the **Search** box in the upper right corner and click  . The search result is displayed.
- The search scope covers only directories with current permissions. You cannot search subdirectories. Search by keywords supports fuzzy match and is case-insensitive. Results of the next page can be searched.

Table 11-46 Manager permission description

Resource Supporting Permission Management	Permission Setting
Alarm	Authorizes the Manager alarm function. You can select **View** to view alarms and **Management** to manage alarms.
Audit	Authorizes the Manager audit log function. You can select **View** to view audit logs and **Management** to manage audit logs.
Dashboard	Authorizes the Manager overview function. You can select **View** to view the cluster overview.
Hosts	Authorizes the node management function. You can select **View** to view node information and **Management** to manage nodes.
Services	Authorizes the service management function. You can select **View** to view service information and **Management** to manage services.
System_cluster_management	Authorizes the MRS cluster management function. You can select **Management** to use the MRS patch management function.
System_configuration	Authorizes the MRS cluster configuration function. You can select **Management** to configure MRS clusters on Manager.
System_task	Authorizes the MRS cluster task function. You can select **Management** to manage periodic tasks of MRS clusters on Manager.

Resource Supporting Permission Management	Permission Setting
Tenant	Authorizes the Manager multi-tenant management function. You can select **Management** to manage multi-tenants.

Table 11-47 HBase permission description

Resource Supporting Permission Management	Permission Setting
SUPER_USER_GROUP	Grants you HBase administrator permissions.
Global	HBase resource type, indicating the whole HBase.
Namespace	HBase resource type, indicating namespace, which is used to store HBase tables. It has the following permissions: ● **Admin** permission to manage the namespace ● **Create**: permission to create HBase tables in the namespace ● **Read**: permission to access the namespace ● **Write**: permission to write data to the namespace ● **Execute**: permission to execute the coprocessor (Endpoint)
Table	HBase resource type, indicating a data table, which is used to store data. It has the following permissions: ● **Admin**: permission to manage a data table ● **Create**: permission to create column families and columns in a data table ● **Read**: permission to read a data table ● **Write**: permission to write data to a data table ● **Execute**: permission to execute the coprocessor (Endpoint)
ColumnFamily	HBase resource type, indicating a column family, which is used to store data. It has the following permissions: ● **Create**: permission to create columns in a column family ● **Read**: permission to read a column family ● **Write**: permission to write data to a column family

Resource Supporting Permission Management	Permission Setting
Qualifier	HBase resource type, indicating a column, which is used to store data. It has the following permissions: ● **Read**: permission to read a column ● **Write**: permission to write data to a column

By default, permissions of an HBase resource type of each level are shared by resource types of sub-levels. However, the **Recursive** option is not selected by default. For example, if **Read** and **Write** permissions are added to the **default** namespace, they are automatically added to the tables, column families, and columns in the namespace. If a child resource is set after the parent resource, the permission of the child resource is the union of the permissions of the parent resource and the current child resource.

Table 11-48 HDFS permission description

Resource Supporting Permission Management	Permission Setting
Folder	HDFS resource type, indicating an HDFS directory, which is used to store files or subdirectories. It has the following permissions: ● **Read**: permission to access the HDFS directory ● **Write**: permission to write data to the HDFS directory ● **Execute**: permission to perform an operation. It must be selected when you add access or write permission.
Files	HDFS resource type, indicating a file in HDFS. It has the following permissions: ● **Read**: permission to access the file ● **Write**: permission to write data to the file ● **Execute**: permission to perform an operation. It must be selected when you add access or write permission.

Permissions of an HDFS directory of each level are not shared by directory types of sub-levels by default. For example, if **Read** and **Execute** permissions are added to the **tmp** directory, you must select **Recursive** for permissions to be added to subdirectories.

Table 11-49 Hive permission description

Resource Supporting Permission Management	Permission Setting
Hive Admin Privilege	Grants you Hive administrator permissions.
Database	Hive resource type, indicating a Hive database, which is used to store Hive tables. It has the following permissions: • **Select**: permission to query the Hive database • **Delete**: permission to perform the deletion operation in the Hive database • **Insert**: permission to perform the insertion operation in the Hive database • **Create**: permission to perform the creation operation in the Hive database
Table	Hive resource type, indicating a Hive table, which is used to store data. It has the following permissions: • **Select**: permission to query the Hive table • **Delete**: permission to perform the deletion operation in the Hive table • **Update**: permission to perform the update operation in the Hive table • **Insert**: permission to perform the insertion operation in the Hive table • **Grant of Select**: permission to grant the **Select** permission to other users using Hive statements • **Grant of Delete**: permission to grant the **Delete** permission to other users using Hive statements • **Grant of Update**: permission to grant the **Update** permission to other users using Hive statements • **Grant of Insert**: permission to grant the **Insert** permission to other users using Hive statements

By default, permissions of a Hive resource type of each level are shared by resource types of sub-levels. However, the **Recursive** option is not selected by default. For example, if **Select** and **Insert** permissions are added to the **default** database, they are automatically added to the tables and columns in the database. If a child resource is set after the parent resource, the permission of the child resource is the union of the permissions of the parent resource and the current child resource.

Table 11-50 Yarn permission description

Resource Supporting Permission Management	Permission Setting
Cluster Admin Operations	Grants you Yarn administrator permissions.
root	Root queue of Yarn. It has the following permissions: • **Submit**: permission to submit jobs in the queue • **Admin**: permission to manage permissions of the current queue
Parent Queue	Yarn resource type, indicating a parent queue containing sub-queues. A root queue is a type of a parent queue. It has the following permissions: • **Submit**: permission to submit jobs in the queue • **Admin**: permission to manage permissions of the current queue
Leaf Queue	Yarn resource type, indicating a leaf queue. It has the following permissions: • **Submit**: permission to submit jobs in the queue • **Admin**: permission to manage permissions of the current queue

By default, permissions of a Yarn resource type of each level are shared by resource types of sub-levels. However, the **Recursive** option is not selected by default. For example, if the **Submit** permission is added to the **root** queue, it is automatically added to the sub-queue. Permissions inherited by sub-queues will not be displayed as selected in the **Permission** table. If a child resource is set after the parent resource, the permission of the child resource is the union of the permissions of the parent resource and the current child resource.

Table 11-51 Hue permission description

Resource Supporting Permission Management	Permission Setting
Storage Policy Admin	Grants you storage policy administrator permissions.

Step 5 Click **OK**. Return to **Manage Role**.

----End

Related Tasks

Modifying a role

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage Role**.

Step 3 In the row of the role to be modified, click **Modify** to modify role information.

NOTE

If you modify permissions assigned by the role, it takes 3 minutes to make new configurations take effect.

Step 4 Click **OK**. The modification is complete.

----End

Deleting a role

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage Role**.

Step 3 In the row of the role to be deleted, click **Delete**.

Step 4 Click **OK**. The role is deleted.

----End

11.13.4 Creating a User Group

Scenario

This section describes how to create user groups and specify their operation permissions on Manager. Management of single or multiple users can be unified in the user groups. After being added to a user group, users can obtain operation permissions owned by the user group.

Manager supports a maximum of 100 user groups.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Managing User Groups**.

Prerequisites

- Administrators have learned service requirements and created roles required by service scenarios.
- You have obtained a cluster with Kerberos authentication enabled or a common cluster with the EIP function enabled.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User Group**.

Step 4 Above the user group list, click **Create User Group**.

Step 5 Input **Group Name** and **Description**.

Group Name is mandatory and contains 3 to 20 characters. Only digits, letters, and underscores (_) are allowed. **Description** is optional.

Step 6 In **Role**, click **Select and Add Role** to select and add specified roles.

If you do not add the roles, the user group you are creating now does not have the permission to use MRS clusters.

Step 7 Click **OK**.

----End

Related Tasks

Modifying a user group

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User Group**.

Step 3 In the row of a user group to be modified, click **Modify**.

NOTE

If you change role permissions assigned to the user group, it takes 3 minutes to make new configurations take effect.

Step 4 Click **OK**. The modification is complete.

----End

Deleting a user group

Step 1 On MRS Manager, click **System**.

Step 2 In the **Permission** area, click **Manage User Group**.

Step 3 In the row of the user group to be deleted, click **Delete**.

Step 4 Click **OK**. The user group is deleted.

----End

11.13.5 Creating a User

Scenario

This section describes how to create users on Manager based on site requirements and specify their operation permissions to meet service requirements.

Up to 1000 users can be created on Manager.

If a new password policy needs to be used for a new user's password, follow instructions in **Modifying a Password Policy** to modify the password policy and then perform the following operations to create a user.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Creating a User**.

Prerequisites

- Administrators have learned service requirements and created roles and role groups required by service scenarios.
- You have obtained a cluster with Kerberos authentication enabled or a common cluster with the EIP function enabled.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 Above the user list, click **Create User**.

Step 5 Configure parameters as prompted and enter a username in **Username**.

NOTE

- A username that differs only in alphabetic case from an existing username is not allowed. For example, if **User1** has been created, you cannot create **user1**.
- When you use the user you created, enter the exactly correct username, which is case-sensitive.
- **Username** is mandatory and contains 3 to 20 characters. Only digits, letters, and underscores (_) are allowed.
- **root**, **omm**, and **ommdba** are reserved system user. Select another username.

Step 6 Set **User Type** to either **Human-machine** or **Machine-machine**.

- **Human-machine** user: used for MRS Manager O&M scenarios and component client operation scenarios. If you select this user type, you need to enter a password and confirm the password in **Password** and **Confirm Password** accordingly.
- **Machine-machine** users: used for MRS application development scenarios. If you select this user type, you do not need to enter a password, because the password is randomly generated.

Step 7 In **User Group**, click **Select and Join User Group** to select user groups and add users to them.

NOTE

- If roles have been added to user groups, the users can be granted with permissions of the roles.
- If you want to grant new users with Hive permissions, add the users to the Hive group.
- If a user needs to manage tenant resources, the user group must be assigned the **Manager_tenant** role and the role corresponding to the tenant.
- Users created on Manager cannot be added to the user group synchronized using the IAM user synchronization function.

Step 8 In **Primary Group**, select a group as the primary group for users to create directories and files. The drop-down list contains all groups selected in **User Group**.

Step 9 In **Assign Rights by Role**, click **Select and Add Role** to add roles for users based on onsite service requirements.

NOTE

- When you create a user, if permissions of a user group that is granted to the user cannot meet service requirements, you can assign other created roles to the user. It takes 3 minutes to make role permissions granted to the new user take effect.
- Adding a role when you create a user can specify the user rights.
- A new user can access web UIs of HDFS, HBase, Yarn, Spark, and Hue even when roles are not assigned to the user.

Step 10 In **Description**, provide description based on onsite service requirements.

Description is optional.

Step 11 Click **OK**.

If a new user is used in the MRS cluster for the first time, for example, used for logging in to MRS Manager or using the cluster client, the password must be changed. For details, see **Changing the Password of an Operation User**.

----End

11.13.6 Modifying User Information

Scenario

This section describes how to modify user information on Manager, including information about the user group, primary group, role, and description.

This operation is supported only in clusters with Kerberos authentication enabled or common clusters with the EIP function enabled.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Modifying User Information**.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 In the row of a user to be modified, click **Modify**.

NOTE

If you change user groups for a user or assign role permissions to a user, it takes 3 minutes to make new configurations take effect.

Step 5 Click **OK**. The modification is complete.

----End

11.13.7 Locking a User

This section describes how to lock users in MRS clusters. A locked user cannot log in to Manager or perform security authentication in the cluster. This operation is supported only in clusters with Kerberos authentication enabled or common clusters with the EIP function enabled.

A locked user can be unlocked by an administrator manually or until the lock duration expires. You can lock a user by using either of the following methods:

- Automatic lock: Set **Number of Password Retries** in **Configure Password Policy**. If user login attempts exceed the parameter value, the user is automatically locked. For details, see **Modifying a Password Policy**.
- Manual lock: The administrator manually locks a user.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Locking a User**.

The following describes how to manually lock a user. **Machine-machine** users cannot be locked.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 In the row of a user you want to lock, click **Lock User**.

Step 5 In the window that is displayed, click **OK** to lock the user.

----End

11.13.8 Unlocking a User

If a user is locked because the number of login attempts exceeds the value of **Number of Password Retries**, or the user is manually locked by the administrator, the administrator can unlock the user on Manager. This operation is supported only in clusters with Kerberos authentication enabled or common clusters with the EIP function enabled.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Unlocking a User**.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 In the row of a user to be unlocked, click **Unlock User**.

Step 5 In the window that is displayed, click **OK** to unlock the user.

----End

11.13.9 Deleting a User

The administrator can delete an MRS cluster user that is not required on MRS Manager. Deleting a user is allowed only in clusters with Kerberos authentication enabled or normal clusters with the EIP function enabled.

NOTE

If you want to create a new user with the same name as user A after deleting user A who has submitted a job on the client or MRS console, you need to delete user A's residual folders when deleting user A. Otherwise, the newly created user A may fail to submit a job.

To delete residual folders, log in to each Core node in the MRS cluster and run the following commands. In the following commands, **$user** indicates the folder named after the username.

cd /srv/BigData/hadoop/data1/nm/localdir/usercache/

rm -rf $user

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Deleting a User**.

Procedure

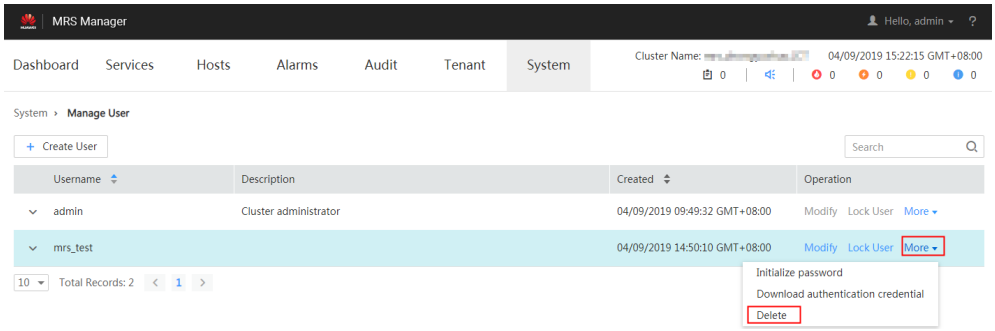
Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 In the row that contains the user to be deleted, choose **More** > **Delete**.

Figure 11-2 Deleting a user



Step 5 Click **OK**.

----End

11.13.10 Changing the Password of an Operation User

Scenario

Passwords of **Human-machine** system users must be regularly changed to ensure MRS cluster security. This section describes how to change passwords on MRS Manager.

If a new password policy needs to be used for the password modified by the user, follow instructions in **Modifying a Password Policy** to modify the password policy and then perform the following operations to modify the password.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Changing a User Password**.

Impact on the System

If you have downloaded a user authentication file, download it again and obtain the keytab file after modifying the password of the MRS cluster user.

Prerequisites

- You have obtained the current password policies from the administrator.
- You have obtained the MRS Manager access address from the administrator.
- You have obtained a cluster with Kerberos authentication enabled or a common cluster with the EIP function enabled.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, move the mouse cursor to in the upper right corner.

On the menu that is displayed, select **Change Password**.

Step 3 Fill in the **Old Password**, **New Password**, and **Confirm Password**. Click **OK**.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.13.11 Initializing the Password of a System User

Scenario

This section describes how to initialize a password on Manager if a user forgets the password or the password of a public account needs to be changed regularly. After password initialization, the user must change the password upon the first login. This operation is supported only in clusters with Kerberos authentication enabled or common clusters with the EIP function enabled.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Initializing a Password**.

Impact on the System

If you have downloaded a user authentication file, download it again and obtain the keytab file after initializing the password of the MRS cluster user.

Initializing the Password of a Human-Machine User

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 Locate the row that contains the user whose password is to be initialized, choose **More** > **Initialize password**, and change the password as prompted.

In the window that is displayed, enter the password of the current administrator account and click **OK**. Then in **Initialize password**, click **OK**.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.

- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

Initializing the Password of a Machine-Machine User

Step 1 Prepare a client based on service conditions and log in to the node with the client installed.

Step 2 Run the following command to switch the user:

sudo su - omm

Step 3 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 Run the following command to log in to the console as user **kadmin/admin**:

NOTE

The default password of user **kadmin/admin** is **KAdmin@123**, which will expire upon your first login. Change the password as prompted and keep the new password secure.

kadmin -p kadmin/admin

Step 6 Run the following command to reset the password of a component running user. This operation takes effect on all servers:

cpw *Component running user name*

For example, **cpw oms/manager**.

For the cluster, the default password complexity requirements are as follows:

- The password must contain 8 to 32 characters.
- The password must contain at least three types of the following: uppercase letters, lowercase letters, digits, spaces, and special characters ('~!@#$ %^&*()-_=+\|[{}];:'",<.>/?).
- The password cannot be the username or the reverse username.

----End

11.13.12 Downloading a User Authentication File

Scenario

When a user develops big data applications and runs them in an MRS cluster that supports Kerberos authentication, the user needs to prepare a **Machine-machine** user authentication file for accessing the MRS cluster. The keytab file in the authentication file can be used for user authentication.

This section describes how to download a **Machine-machine** user authentication file and export the keytab file on Manager. This operation is supported only in clusters with Kerberos authentication enabled or common clusters with the EIP function enabled.

NOTE

Before downloading a **Human-machine** user authentication file, change the password for the user on MRS Manager to make the initial password set by the administrator invalid. Otherwise, the exported keytab file cannot be used. For details, see **Changing the Password of an Operation User**.

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Exporting an Authentication Credential File**.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 In the **Permission** area, click **Manage User**.

Step 4 In the row of a user for whom you want to export the keytab file, choose **More** > **Download authentication credential** to download the authentication file. After the file is automatically generated, save it to a specified path and keep it secure.

Step 5 Open the authentication file with a decompression program.

- **user.keytab** indicates a user keytab file used for user authentication.
- **krb5.conf** indicates the configuration file of the authentication server. The application connects to the authentication server according to this configuration file information when authenticating users.

----End

11.13.13 Modifying a Password Policy

Scenario

NOTICE

Because password policies are critical to the user management security, modify them based on service security requirements. Otherwise, security risks may be incurred.

This section describes how to set password and user login security rules as well as user lock rules. Password policies set on MRS Manager take effect for **Human-machine** users only, because the passwords of **Machine-machine** users are randomly generated. This operation is supported only in clusters with Kerberos authentication enabled or common clusters with the EIP function enabled.

If a new password policy needs to be used for a new user's password or the password modified by the user, perform the following operations to modify the

password policy first, and then follow instructions in **Creating a User** or **Changing the Password of an Operation User**.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Configuring Password Policies**.

Procedure

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 On MRS Manager, click **System**.

Step 3 Click **Configure Password Policy**.

Step 4 Modify password policies as prompted. For parameter details, see **Table 11-52**.

Table 11-52 Password policy parameter description

Parameter	Description
Minimum Password Length	Indicates the minimum number of characters a password contains. The value ranges from 8 to 32. The default value is **8**.
Number of Character Types	Indicates the minimum number of character types a password contains. The character types include uppercase letters, lowercase letters, digits, spaces, and special characters (~`!?,.:;-_'(){}[]/<>@#$ %^&*+|\=). The value can be **3** or **4**. The default value **3** indicates that the password must contain at least three types of the following characters: uppercase letters, lowercase letters, digits, special characters, and spaces.
Password Validity Period (days)	Indicates the validity period (days) of a password. The value ranges from 0 to 90. Value **0** means that the password is permanently valid. The default value is **90**.

Parameter	Description
Password Expiration Notification Days	Indicates the number of days to notify password expiration in advance. After the value is set, if the difference between the cluster time and the password expiration time is smaller than this value, the user receives password expiration notifications. When a user logs in to MRS Manager, a message is displayed, indicating that the password is about to expire and asking the user whether to change the password. The value ranges from **0** to *X* (*X* must be set to the half of the password validity period and rounded down). Value **0** indicates that no notification is sent. The default value is **5**.
Interval of Resetting Authentication Failure Count (min)	Indicates the interval (minutes) of retaining incorrect password attempts. The value ranges from 0 to 1440. Value **0** indicates that the number of incorrect password attempts are permanently retained and value **1440** indicates that the number of incorrect password attempts are retained for one day. The default value is **5**.
Number of Password Retries	Indicates the number of consecutive wrong passwords allowed before the system locks the user. The value ranges from 3 to 30. The default value is **5**.
Account Lock Duration (min)	Indicates the time period for which a user is locked when the user lockout conditions are met. The value ranges from 5 to 120. The default value is **5**.

----End

11.13.14 Configuring Cross-Cluster Mutual Trust Relationships

Scenario

If cluster A needs to access the resources of cluster B, the mutual trust relationship must be configured between these two clusters.

If no trust relationship is configured, resources of a cluster are available only for users in this cluster. MRS automatically assigns a unique **domain name** for each cluster to define the scope of resources for users.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**.

Impact on the System

- After cross-cluster mutual trust is configured, resources of a cluster become available for users in other cluster. User permission in the clusters must be regularly checked based on service and security requirements.
- After cross-cluster mutual trust is configured, the KrbServer service needs to be restarted and the cluster becomes unavailable during the restart.
- After cross-cluster mutual trust is configured, internal users **krbtgt/***Local cluster domain name*@*External cluster domain name* and **krbtgt/***External cluster domain name*@*Local cluster domain name* are added to the two clusters. The internal users cannot be deleted.

Procedure

Step 1 On the MRS management console, query all security groups of the two clusters.

- If the security groups of the two clusters are the same, go to **Step 3**.
- If the security groups of the two clusters are different, go to **Step 2**.

Step 2 On the VPC management console, add rules for each security group.

Set **Protocol** to **ANY**, **Transfer Direction** to **Inbound**,

and **Source** to **Security Group**. The source is the security group of the peer cluster.

- For cluster A, add inbound rules to the security group, set **Source** to the security groups of cluster B (the peer cluster of cluster A).
- For cluster B, add inbound rules to the security group, set **Source** to the security groups of cluster A (the peer cluster of cluster B).

NOTE

For a common cluster with Kerberos authentication disabled, perform step **Step 1** to **Step 2** to configure cross-cluster mutual trust. For a security cluster with Kerberos authentication enabled, after completing the preceding steps, proceed to the following steps for configuration.

Step 3 Log in to MRS Manager of the two clusters separately. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**. Click **Service** and check whether the **Health Status** of all components is **Good**.

- If yes, go to **Step 4**.
- If no, contact technical support personnel for troubleshooting.

Step 4 Query configuration information.

1. On MRS Manager of the two clusters, choose **Services** > **KrbServer** > **Instance**. Query the **OM IP Address** of the two KerberosServer hosts.
2. Click **Service Configuration**. Set **Type** to **All**. Choose **KerberosServer** > **Port** in the navigation tree on the left. Query the value of **kdc_ports**. The default value is **21732**.

3. Click **Realm** and query the value of **default_realm**.

Step 5 On MRS Manager of either cluster, modify the **peer_realms** parameter.

Table 11-53 Parameter description

Parameter	Description
realm_name	Domain name of the mutual-trust cluster, that is, the value of **default_realm** obtained in step 4.
ip_port	KDC address of the peer cluster. Format: *IP address of a KerberosServer node in the peer cluster:kdc_port* The addresses of the two KerberosServer nodes are separated by a comma. For example, if the IP addresses of the KerberosServer nodes are 10.0.0.1 and 10.0.0.2 respectively, the value of this parameter is **10.0.0.1:21732,10.0.0.2:21732**.

NOTE

- To deploy trust relationships with multiple clusters, click  to add items and specify relevant parameters. To delete an item, click  .
- A cluster can have trust relationships with a maximum of 16 clusters. By default, no trust relationship exists between different clusters that are trusted by a local cluster.

Step 6 Click **Save Configuration**. In the dialog box that is displayed, select **Restart the affected services or instances** and click **OK**. If you do not select **Restart the affected services or instances**, manually restart the affected services or instances.

After **Operation successful** is displayed, click **Finish**.

Step 7 Exit MRS Manager and log in to it again. If the login is successful, the configurations are valid.

Step 8 Log in to MRS Manager of the other cluster and repeat step **Step 5** to **Step 7**.

----End

Follow-up Operations

After cross-cluster mutual trust is configured, the service configuration parameters are modified on MRS Manager and the service is restarted. Therefore, you need to prepare the client configuration file again and update the client.

Scenario 1:

Cluster A and cluster B (peer cluster and mutually trusted cluster) are the same type, for example, analysis cluster or streaming cluster. Follow instructions in **Updating a Client (Versions Earlier Than 3.x)** to update the client configuration files of cluster A and B respectively.

- Update the client configuration file of cluster A.
- Update the client configuration file of cluster B.

Scenario 2:

Cluster A and cluster B (peer cluster and mutually trusted cluster) are the different type. Perform the following steps to update the configuration files.

- Update the client configuration file of cluster A to cluster B.
- Update the client configuration file of cluster B to cluster A.
- Update the client configuration file of cluster A.
- Update the client configuration file of cluster B.

Step 1 Log in to MRS Manager of cluster A.

Step 2 Click **Services**, and then **Download Client**.

Step 3 Set **Client Type** to **Only configuration files**.

Step 4 Set **Download to** to **Remote host**.

Step 5 Set **Host IP Address** to the IP address of the active Master node of cluster B, **Host Port** to 22, and **Save Path** to **/tmp**.

- If the default port **22** for logging in to cluster B using SSH is changed, set **Host Port** to a new port.
- The value of **Save Path** contains a maximum of 256 characters.

Step 6 Set **Login User** to **root**.

If another user is used, ensure that the user has permissions to read, write, and execute the save path.

Step 7 Select **Password** or **SSH Private Key** for **Login Mode**.

- **Password**: Enter the password of user **root** set during cluster creation.
- **SSH Private Key**: Select and upload the key file used for creating the cluster.

Step 8 Click **OK** to generate a client file.

If the following information is displayed, the client file is saved. Click **Close**.

```
Client files downloaded to the remote host successfully.
```

If the following information is displayed, check the username, password, and security group configurations of the remote host. Ensure that the username and password are correct and an inbound rule of the SSH (22) port has been added to the security group of the remote host. And then, go to **Step 2** to download the client again.

```
Failed to connect to the server. Please check the network connection or parameter settings.
```

Step 9 Log in to the ECS of cluster B using VNC. For details, see

Step 10 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 11 Run the following command to update the client configuration of cluster A to cluster B:

sh refreshConfig.sh *Client installation directory Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/Bigdata/client /tmp/MRS_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully.

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

NOTE

You can also refer to method 2 in **Updating a Client (Versions Earlier Than 3.x)** to perform operations in **Step 1** to **Step 11**.

Step 12 Repeat step **Step 1** to **Step 11** to update the client configuration file of cluster B to cluster A.

Step 13 Follow instructions in **Updating a Client (Versions Earlier Than 3.x)** to update the client configuration file of the local cluster.

- Update the client configuration file of cluster A.
- Update the client configuration file of cluster B.

----End

11.13.15 Configuring Users to Access Resources of a Trusted Cluster

Scenario

After cross-cluster mutual trust is configured, permission must be configured for users in the local cluster, so that the users can access the same resources in the peer cluster as the users in the peer cluster.

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

For clusters of **MRS 3.*x*** or later, see **Configuring Cross-Manager Cluster Mutual Trust Relationships**.

Prerequisites

The mutual trust relationship has been configured between two clusters (clusters A and B). The clients of the clusters have been updated.

Procedure

Step 1 Log in to MRS Manager of cluster A and choose **System** > **Manage User**. Check whether cluster A has accounts that are the same as those of cluster B.

- If yes, go to **Step 2**.
- If no, go to **Step 3**.

Step 2 Click on the left side of the username to unfold the detailed user information. Check whether the user group and role to which the user belongs meet the service requirements.

For example, user **admin** of cluster A has the permission to access and create files in the **/tmp** directory of cluster A. Then go to **Step 4**.

Step 3 Create the accounts in cluster A and bind the accounts to the user group and roles required by the services. Then go to **Step 4**.

Step 4 Choose **Service** > **HDFS** > **Instance**. Query the **OM IP Address** of **NameNode (Active)**.

Step 5 Log in to the client of cluster B.

For example, if you have updated the client on the Master2 node, log in to the Master2 node to use the client. For details, see **Using an MRS Client**.

Step 6 Run the following command to access the **/tmp** directory of cluster A.

hdfs dfs -ls hdfs://192.168.6.159:9820/tmp

In the preceding command, **192.168.6.159** is the IP address of the active NameNode of cluster A; **9820** is the default port for communication between the client and the NameNode.

Step 7 Run the following command to create a file in the **/tmp** directory of cluster A:

hdfs dfs -touchz hdfs://192.168.6.159:9820/tmp/mrstest.txt

If you can query the **mrstest.txt** file in the **/tmp** directory of cluster A, the cross-cluster mutual trust is configured successfully.

----End

11.13.16 Configuring Fine-Grained Permissions for MRS Multi-User Access to OBS

When fine-grained permission control is enabled, you can configure OBS access permissions to implement access control on directories in OBS file systems.

This function enables you to control MRS users' access to OBS resources. For example, if you allow user group A to only access log files in a specified OBS file system, perform the following operations:

1. Configure an agency with OBS access permissions for an MRS cluster so that OBS can be accessed using the temporary AK/SK automatically obtained by the ECS. This prevents the AK/SK from being exposed in the configuration file.
2. Create a policy on the IAM console to allow access to log files in a specified OBS file system, and create an agency bound to the policy permission.
3. In the MRS cluster, bind the new agency to user group A so that user group A only has the permission to access log files in the specified OBS file system.

In the following scenarios, the username used for submitting jobs is an internal username so that MRS multi-user access to OBS is not supported.

- For spark-beeline, the internal username used for submitting jobs is **spark** in a security cluster and **omm** in a normal cluster.

- For the HBase shell, the internal username used for submitting jobs is **hbase** in a security cluster and **omm** in a normal cluster.
- For Presto, the internal username used for submitting jobs in the security cluster is **omm** or **hive**, and that in the normal cluster is **omm**. (Choose **Components** > **Presto** > **Service Configuration**. Change **Basic** to **All** in the parameter type drop-down box.) Then, search for and change the value of **hive.hdfs.impersonation.enabled** to **true** to enable MRS multi-user to access OBS with fine-grained permissions.

Prerequisites

- Fine-grained permission control has been enabled. For details about permissions management, see **Creating an MRS User**.
- You have a basic knowledge of and OBS fine-grained policies.

Step 1: Configuring an Agency with OBS Access Permission for a Cluster

Step 1 Follow instructions in **Configuring a Storage-Compute Decoupled Cluster (Agency)** to configure an agency with OBS access permissions.

The agency takes effect for all users (including internal users) and user groups in the cluster. To control the permissions of users and user groups in the cluster to access OBS, perform the following operations.

----End

Step 2: Creating a Policy and an Agency on IAM

Create policies with different access permissions and bind the policies to the agency. For details, see **Creating a Policy and an Agency on IAM**.

Step 3: Configuring OBS Permission Control Mappings on the MRS Cluster Details Page

Step 1 On the MRS management console, choose **Clusters** > **Active Clusters** and click the cluster name.

Step 2 In the **Basic Information** area on the **Dashboard** tab page, click **Manage** next to **OBS Permission Control**.

Step 3 Click **Add Mapping** and set parameters according to **Table 11-54**.

Table 11-54 OBS permission control parameters

Parameter	Description
IAM Agency	Select the agency created in **Step 2**.

Parameter	Description
Type	• **User**: User-level mapping • **Group**: User group-level mapping **NOTE** • User-level mapping takes priority over user group-level mapping. If you select **Group**, you are advised to enter the primary group name in **MRS User (User Group)**. • Do not use the same username (user group) for multiple mapping records.
MRS User (User Group)	Use commas (,) to separate multiple names of users or user groups. **NOTE** • If OBS permission control is not configured for a user and no AK and SK are configured, the permission in **MRS_ECS_DEFAULT_AGENCY** will be used for accessing OBS. You are advised not to bind the internal user of a component to an agency. • If you need to configure an agency for the internal user of a component when submitting a job in the following scenarios, the requirements are as follows: – To control permissions on spark-beeline operations, set the username to **spark** for a security cluster and **omm** for a normal cluster. – To control permissions on HBase shell operations, set the username to **hbase** for a security cluster and **omm** for a normal cluster. – To control permissions on Presto, set the username to **omm**, **hive**, and the username used for logging in to the client for a security cluster and **omm** and the username used for logging in to the client for a normal cluster. – If you want to use Hive to create tables in beeline mode, set the username to the internal user **hive**.

Step 4 Click **OK**.

Step 5 Select **I agree to authorize the trust relationships between MRS Users (Groups) and IAM agencies**, and click **OK**. The mapping between the MRS user and OBS permission is added.

If appears next to **OBS Permission Control** on the **Dashboard** tab page or the mapping table has been updated for OBS permission control, the mapping takes effect. It takes about 1 minute to for the mapping to take effect.

In the **Operation** column of the mapping list, you can edit or delete the added mapping.

NOTE

- If OBS permission control is not configured for a user and no AK and SK are configured, the permissions owned by the agency configured for the cluster in the **Object Storage Service (OBS)** project will be used to access OBS.
- Regardless of whether OBS permission control is configured, AK/SK permission is used for accessing OBS once it is configured.
- Security Administrator permission is required to modify, create, or delete a mapping.
- To enable mapping changes to take effect in spark-line, hive beeline and Presto respectively, you need to restart Spark, exit beeline and enter again, and restart Presto respectively.

----End

Component Access to OBS When OBS Permission Control Is Enabled

Step 1 Log in to any node in a cluster as user **root** using the password set during cluster creation.

Step 2 Set environment variables (In MRS 3.x and later versions, the default installation path of the client is /opt/Bigdata/client. In MRS 3.x and earlier versions, the default installation path is /opt/client. For details, see the actual situation.).

source /opt/Bigdata/client/bigdata_env

Step 3 If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If the Kerberos authentication is disabled for the current cluster, skip this step:

kinit *MRS cluster user*

Example: **kinit admin**

Step 4 If the Kerberos authentication is disabled for the current cluster, run the following commands to log in. Note that you should create a user that belongs to the **supergroup** group by referring to **Creating a User** and replace *XXXX* with the username:

mkdir /home/XXXX

chown XXXX /home/XXXX

su - XXXX

Step 5 Access OBS. You do not need to configure the AK, SK, and endpoint. The OBS path format is **obs://buck_name/*XXX***.

Example: **hadoop fs -ls "obs://obs-example/job/hadoop-mapreduce-examples-3.1.2.jar"**

NOTE

- If you want to use **hadoop fs** to delete files on OBS, use **hadoop fs -rm -skipTrash** to delete the files.
- If data import is not involved when a table is created using spark-sql and spark-beeline, OBS will not be accessed. That is, if you create a table in an OBS directory on which you do not have permission, the **CREATE TABLE** operation will still be successful, but the error message "**403 AccessDeniedException**" is displayed when you insert data.

----End

Creating a Policy and an Agency on IAM

Step 1 Create a policy on IAM.

1. Log in to the IAM console.
2. Choose **Permissions**. On the displayed page, click **Create Custom Policy**.
3. Set parameters according to **Table 11-55**. Obtain the customized OBS policy samples that are frequently used by referring to .

Table 11-55 Policy parameters

Parameter	Description
Policy Name	Only letters, digits, spaces, and special characters (-_.,) are allowed.
Scope	Select **Global services**, because OBS is a global service.
Policy View	Select **Visual editor**.

Parameter	Description
Policy Content	1. **Allow**: Select **Allow**. 2. **Select service**: Select **Object Storage Service (OBS)**. 3. **Select action**: Select **WriteOnly**, **ReadOnly**, and **ListOnly**. 4. **Specific resources**: a. Set **object** to **Specify resource path**, click **Add Resource Path**, and enter *obs_bucket_name*/**tmp/** and *obs_bucket_name*/**tmp/***. The **/tmp** directory is used as an example. If you need to add permissions for other directories, perform the following steps to add the directories and resource paths of all objects in the directories. b. Set **bucket** to **Specify resource path**, click **Add Resource Path**, and enter *obs_bucket_name*. 5. (Optional) Add request condition, which does not need to be added currently.
Description	(Optional) Brief description about the policy.

NOTE

If the data write operation of each component is implemented in **rename** mode, the permission to delete objects must be configured when data is written.

4. Click **OK** to save the policy.

Step 2 Create an agency on IAM.

1. Log in to the IAM console.
2. Choose **Agencies**. On the displayed page, click **Create Agency**.
3. Set parameters according to **Table 11-56**.

Table 11-56 Agency parameters

Parameter	Description
Agency Name	Only letters, digits, spaces, and special characters (-_.,) are allowed.

Parameter	Description
Agency Type	Select **Common account**.
Delegated Account	Enter your cloud account, that is, the account you register using your mobile phone number. It cannot be a federated user or an IAM user created using your cloud account.
Validity Period	Set this parameter as required.
Description	(Optional) Brief description about the agency.
Permissions	1. In the **Project [Region]** column, locate the row where **OBS** is, click **Attach Policy**. 2. Select the policy created in **Step 1** to display it in **Selected Policies**. 3. Click **OK**.

4. Click **OK** to save the agency.

   NOTE

   If you modify an agency and policies bound to it after using the agency to access OBS, the modification will take effect within 15 minutes.

----End

11.14 Patch Operation Guide

11.14.1 Patch Operation Guide for Versions

If you obtain patch information from the following sources, upgrade the patch according to actual requirements.

- You obtain information about the patch released by MRS from a message pushed by the message center service.
- You obtain information about the patch by accessing the cluster and viewing patch information.

Preparing for Patch Installation

- Follow instructions in **Performing a Health Check** to check cluster status. If the cluster health status is normal, install a patch.
- You need to confirm the target patch to be installed according to the patch information in the patch content.

Installing a Patch

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patch Information** page, click **Install** in the **Operation** column to install the target patch.

NOTE

- For details about rolling patch operations, see **Supporting Rolling Patches**.
- For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

Uninstalling a Patch

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters > Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patch Information** page, click **Uninstall** in the **Operation** column to uninstall the target patch.

NOTE

- For details about rolling patch operations, see **Supporting Rolling Patches**.
- For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

11.14.2 Supporting Rolling Patches

The rolling patch function indicates that patches are installed or uninstalled for one or more services in a cluster by performing a rolling service restart (restarting services or instances in batches), without interrupting the services or within a minimized service interruption interval. Services in a cluster are divided into the following three types based on whether they support rolling patch:

- Services supporting rolling patch installation or uninstallation: All businesses or part of them (varying depending on different services) of the services are not interrupted during patch installation or uninstallation.
- Services not supporting rolling patch installation or uninstallation: Businesses of the services are interrupted during patch installation or uninstallation.
- Services with some roles supporting rolling patch installation or uninstallation: Some businesses of the services are not interrupted during patch installation or uninstallation.

Table 11-57 provides services and instances that support or do not support rolling restart in the MRS cluster.

Table 11-57 Services and instances that support or do not support rolling restart

Service	Instance	Whether to Support Rolling Restart
HDFS	NameNode	Yes
	ZKFC	
	JournalNode	
	HttpFS	
	DataNode	
Yarn	ResourceManager	Yes
	NodeManager	
Hive	MetaStore	Yes
	WebHCat	
	HiveServer	
MapReduce	JobHistoryServer	Yes
HBase	HMaster	Yes
	RegionServer	
	ThriftServer	
	RESTServer	
Spark	JobHistory	Yes
	JDBCServer	
	SparkResource	No
Hue	Hue	No
Tez	TezUI	No
Loader	Sqoop	No
ZooKeeper	QuorumPeer	Yes
Kafka	Broker	Yes
	MirrorMaker	No
Flume	Flume	Yes
	MonitorServer	
Storm	Nimbus	Yes
	UI	
	Supervisor	

Service	Instance	Whether to Support Rolling Restart
	LogViewer	

Installing a Patch

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters** > **Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patch Information** page, click **Install** in the **Operation** column.

Step 4 On the **Warning** page, enable or disable **Rolling Patch**.

NOTE

- Enabling the rolling patch installation function: Services are not stopped before patch installation, and rolling service restart is performed after the patch installation. This minimizes the impact on cluster services but takes more time than common patch installation.
- Disabling the rolling patch uninstallation function: All services are stopped before patch uninstallation, and all services are restarted after the patch uninstallation. This temporarily interrupts the cluster and the services but takes less time than rolling patch uninstallation.
- The rolling patch installation function is not available in clusters with less than two Master nodes and three Core nodes.

Step 5 Click **OK** to install the target patch.

Step 6 View the patch installation progress.

1. Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.
2. Choose **System** > **Manage Patch**. On the **Manage Patch** page, you can view the patch installation progress.

NOTE

For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

Uninstalling a Patch

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters** > **Active Clusters** and click the name of the cluster to be queried to enter the page displaying the cluster's basic information.

Step 3 On the **Patch Information** page, click **Uninstall** in the **Operation** column.

Step 4 On the **Warning** page, enable or disable **Rolling Patch**.

NOTE

- Enabling the rolling patch uninstallation function: Services are not stopped before patch uninstallation, and rolling service restart is performed after the patch uninstallation. This minimizes the impact on cluster services but takes more time than common patch uninstallation.
- Disabling the rolling patch uninstallation function: All services are stopped before patch uninstallation, and all services are restarted after the patch uninstallation. This temporarily interrupts the cluster and the services but takes less time than rolling patch uninstallation.
- The rolling patch uninstallation function is not available in clusters with less than two Master nodes and three Core nodes.

Step 5 Click **OK** to uninstall the target patch.

Step 6 View the patch uninstallation progress.

1. Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.
2. Choose **System** > **Manage Patch**. On the **Manage Patch** page, you can view the patch uninstallation progress.

NOTE

For the isolated host nodes in the cluster, follow instructions in **Restoring Patches for the Isolated Hosts** to restore the patch.

----End

11.15 Restoring Patches for the Isolated Hosts

If some hosts are isolated in a cluster, perform the following operations to restore patches for these isolated hosts after patch installation on other hosts in the cluster. After patch restoration, versions of the isolated host nodes are consistent with those are not isolated.

Step 1 Access MRS Manager. For details, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.

Step 2 Choose **System** > **Manage Patch**. The **Manage Patch** page is displayed.

Step 3 In the **Operation** column, click **View Details**.

Step 4 On the patch details page, select host nodes whose **Status** is **Isolated**.

Step 5 Click **Select and Restore** to restore the isolated host nodes.

----End

11.16 Rolling Restart

After modifying the configuration items of a big data component, you need to restart the corresponding service to make new configurations take effect. If you use a normal restart mode, all services or instances are restarted concurrently, which may cause service interruption. To ensure that services are not affected during service restart, you can restart services or instances in batches by rolling

restart. For instances in active/standby mode, a standby instance is restarted first and then an active instance is restarted. Rolling restart takes longer than normal restart.

Table 11-58 provides services and instances that support or do not support rolling restart in the MRS cluster.

Table 11-58 Services and instances that support or do not support rolling restart

Service	Instance	Whether to Support Rolling Restart
HDFS	NameNode	Yes
	ZKFC	
	JournalNode	
	HttpFS	
	DataNode	
Yarn	ResourceManager	Yes
	NodeManager	
Hive	MetaStore	Yes
	WebHCat	
	HiveServer	
MapReduce	JobHistoryServer	Yes
HBase	HMaster	Yes
	RegionServer	
	ThriftServer	
	RESTServer	
Spark	JobHistory	Yes
	JDBCServer	
	SparkResource	No
Hue	Hue	No
Tez	TezUI	No
Loader	Sqoop	No
ZooKeeper	Quorumpeer	Yes
Kafka	Broker	Yes
	MirrorMaker	No
Flume	Flume	Yes

Service	Instance	Whether to Support Rolling Restart
	MonitorServer	
Storm	Nimbus	Yes
	UI	
	Supervisor	
	Logviewer	

Restrictions

- Perform a rolling restart during off-peak hours.
  - Otherwise, a rolling restart failure may occur. For example, if the throughput of Kafka is high (over 100 MB/s) during the Kafka rolling restart, the Kafka rolling restart may fail.
  - For example, if the requests per second of each RegionServer on the native interface exceed 10,000 during the HBase rolling restart, you need to increase the number of handles to prevent a RegionServer restart failure caused by heavy loads during the restart.
- Before the restart, check the number of current requests of HBase. If requests of each RegionServer on the native interface exceed 10,000, increase the number of handles to prevent a failure.
- If the number of Core nodes in a cluster is less than six, services may be affected for a short period of time.
- Preferentially perform a rolling instance or service restart and select **Only restart instances whose configurations have expired**.

Performing a Rolling Service Restart

Step 1 On MRS Manager, click **Services** and select a service for which you want to perform a rolling restart.

Step 2 On the **Service Status** tab page, click **More** and select **Perform Rolling Service Restart**.

Step 3 After you enter the administrator password, the **Perform Rolling Service Restart** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the service.

Step 4 After the rolling restart task is complete, click **Finish**.

----End

Performing a Rolling Instance Restart

Step 1 On MRS Manager, click **Services** and select a service for which you want to perform a rolling restart.

Step 2 On the **Instance** tab page, select the instance to be restarted. Click **More** and select **Perform Rolling Instance Restart**.

Step 3 After you enter the administrator password, the **Perform Rolling Instance Restart** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the instance.

Step 4 After the rolling restart task is complete, click **Finish**.

----End

Perform a Rolling Cluster Restart

Step 1 On MRS Manager, click **Services**. The **Services** page is displayed.

Step 2 Click **More** and select **Perform Rolling Cluster Restart**.

Step 3 After you enter the administrator password, the **Perform Rolling Cluster Restart** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the cluster.

Step 4 After the rolling restart task is complete, click **Finish**.

----End

Rolling Restart Parameter Description

Table 11-59 describes rolling restart parameters.

Table 11-59 Rolling restart parameter description

Parameter	Description
Only restart instances whose configurations have expired	Specifies whether to restart only the modified instances in a cluster.
Data Node Instances to Be Batch Restarted	Specifies the number of instances that are restarted in each batch when the batch rolling restart strategy is used. The default value is **1**. The value ranges from 1 to 20. This parameter is valid only for data nodes.
Batch Interval	Specifies the interval between two batches of instances for rolling restart. The default value is **0**. The value ranges from 0 to 2147483647. The unit is second. Note: Setting the batch interval parameter can increase the stability of the big data component process during the rolling restart. You are advised to set this parameter to a non-default value, for example, 10.
Batch Fault Tolerance Threshold	Specifies the tolerance times when the rolling restart of instances fails to be executed in batches. The default value is **0**, which indicates that the rolling restart task ends after any batch of instances fails to be restarted. The value ranges from 0 to 214748364.

Procedure in a Typical Scenario

Step 1 On MRS Manager, click **Services** and select HBase. The HBase service page is displayed.

Step 2 Click the **Service Configuration** tab, and modify an HBase parameter. After the following dialog box is displayed, click **OK** to save the configurations.

NOTE

Do not select **Restart the affected services or instances**. This option indicates a normal restart. If you select this option, all services or instances will be restarted, which may cause service interruption.

Step 3 After saving the configurations, click **Finish**.

Step 4 Click the **Service Status** tab.

Step 5 On the **Service Status** tab page, click **More** and select **Perform Rolling Service Restart**.

Step 6 After you enter the administrator password, the **Perform Rolling Service Restart** page is displayed. Select **Only restart instances whose configurations have expired** and click **OK** to perform rolling restart for the service.

Step 7 After the rolling restart task is complete, click **Finish**.

----End

12 MRS Cluster Component Operation Guide

12.1 Using Alluxio

12.1.1 Configuring an Underlying Storage System

If you want to use a unified client API and a global namespace to access persistent storage systems including HDFS and OBS to separate computing from storage, you can configure the underlying storage system of Alluxio on MRS Manager. After a cluster is created, the default underlying storage address is **hdfs://hacluster/**, that is, the HDFS root directory is mapped to Alluxio.

Prerequisites

- Alluxio has been installed in a cluster.
- The password of user **admin** has been obtained. The password of user **admin** is specified by the user during MRS cluster creation.

Configuring HDFS as the Underlying File System of Alluxio

NOTE

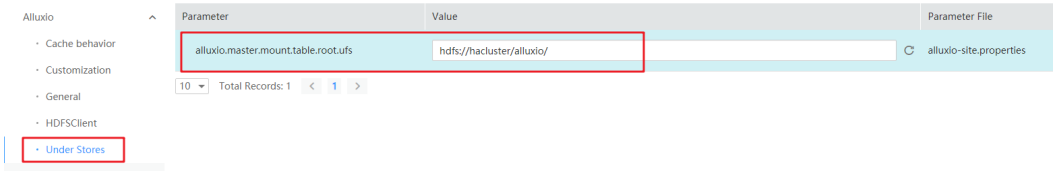
Security clusters with Kerberos authentication enabled do not support this function.

Step 1 Go to the **All Configurations** page of Alluxio. See **Modifying Cluster Service Configuration Parameters**.

Step 2 In the left pane, choose **Alluxio** > **Under Stores**, and modify the value of **alluxio.master.mount.table.root.ufs** to **hdfs://hacluster***/XXX/*.

For example, if you want to use *HDFS root directory*/**alluxio/** as the root directory of Alluxio, modify the value of **alluxio.master.mount.table.root.ufs** to **hdfs://hacluster/alluxio/**.

Figure 12-1 Configuring HDFS as the underlying file system of Alluxio



Step 3 Click **Save Configuration**. In the displayed dialog box, select **Restart the affected services or instances**.

Step 4 Click **OK** to restart Alluxio.

----End

12.1.2 Accessing Alluxio Using a Data Application

The port number used for accessing the Alluxio file system is 19998, and the access address is **alluxio://***<Master node IP address of Alluxio>***:19998/***<PATH>*. This section uses examples to describe how to access the Alluxio file system using data applications (Spark, Hive, Hadoop MapReduce, and Presto).

Using Alluxio as the Input and Output of a Spark Application

Step 1 Log in to the Master node in a cluster as user **root** using the password set during cluster creation.

Step 2 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:

kinit *MRS cluster user*

Example: **kinit admin**

Step 4 Prepare an input file and copy local data to the Alluxio file system.

For example, prepare the input file **test_input.txt** in the local **/home** directory, and run the following command to save the **test_input.txt** file to Alluxio:

alluxio fs copyFromLocal /home/test_input.txt /input

Step 5 Run the following commands to start **spark-shell**:

spark-shell

Step 6 Run the following commands in spark-shell:

val s = sc.textFile("alluxio://<*Name of the Alluxio node*>:19998/input")

val double = s.map(line => line + line)

double.saveAsTextFile("alluxio://<*Name of the Alluxio node*>:19998/output")

NOTE

Replace ***Name of the Alluxio node*****:19998** with the actual node name and port numbers of all nodes where the AlluxioMaster instance is deployed. Use commas (,) to separate the node name and port number, for example, **node-ana-coremspb.mrs-m0va.com:19998,node-master2kiww.mrs-m0va.com:19998,node-master1cqwv.mrs-m0va.com:19998**.

Step 7 Press **Ctrl+C** to exit spark-shell.

Step 8 Run the **alluxio fs ls /** command to check whether the output directory **/output** containing double content of the input file exists in the root directory of Alluxio.

----End

Creating a Hive Table on Alluxio

Step 1 Log in to the Master node in a cluster as user **root** using the password set during cluster creation.

Step 2 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:

kinit *MRS cluster user*

Example: **kinit admin**

Step 4 Prepare an input file. For example, prepare the **hive_load.txt** input file in the local **/home** directory. The file content is as follows:

```
1, Alice, company A
2, Bob, company B
```

Step 5 Run the following command to import the **hive_load.txt** file to Alluxio:

alluxio fs copyFromLocal /home/hive_load.txt /hive_input

Step 6 Run the following command to start the Hive beeline:

beeline

Step 7 Run the following commands in beeline to create a table based on the input file in Alluxio:

CREATE TABLE u_user(id INT, name STRING, company STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;

LOAD DATA INPATH 'alluxio://<*Name of the Alluxio node*>:19998/hive_input' INTO TABLE u_user;

NOTE

Replace ***Name of the Alluxio node*****:19998** with the actual node name and port numbers of all nodes where the AlluxioMaster instance is deployed. Use commas (,) to separate the node name and port number, for example, **node-ana-coremspb.mrs-m0va.com:19998,node-master2kiww.mrs-m0va.com:19998,node-master1cqwv.mrs-m0va.com:19998**.

Step 8 Run the following command to view the created table:

select * from u_user;

----End

Running Hadoop Wordcount in Alluxio

Step 1 Log in to the Master node in a cluster as user **root** using the password set during cluster creation.

Step 2 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:

kinit *MRS cluster user*

Example: **kinit admin**

Step 4 Prepare an input file and copy local data to the Alluxio file system.

For example, prepare the input file **test_input.txt** in the local **/home** directory, and run the following command to save the **test_input.txt** file to Alluxio:

alluxio fs copyFromLocal /home/test_input.txt /input

Step 5 Run the following command to execute the wordcount job:

yarn jar /opt/share/hadoop-mapreduce-examples-<*Hadoop version*>-mrs-<*MRS cluster version*>/hadoop-mapreduce-examples-<*Hadoop version*>-mrs-<*MRS cluster version*>.jar wordcount alluxio://<*Name of the Alluxio node*>:19998/input alluxio://<*Name of the Alluxio node*>:19998/output

NOTE

- Replace **<*Hadoop version*>** with the actual one.
- Replace **<*MRS cluster version*>** with the major version of MRS. For example, for a cluster of MRS 1.9.2, mrs-1.9.0 is used.
- Replace ***Name of the Alluxio node*>:19998** with the actual node name and port numbers of all nodes where the AlluxioMaster instance is deployed. Use commas (,) to separate the node name and port number, for example, **node-ana-coremspb.mrs-m0va.com:19998,node-master2kiww.mrs-m0va.com:19998,node-master1cqwv.mrs-m0va.com:19998**.

Step 6 Run the **alluxio fs ls /** command to check whether the output directory **/output** containing the wordcount result exists in the root directory of Alluxio.

----End

Using Presto to Query Tables in Alluxio

Step 1 Log in to the Master node in a cluster as user **root** using the password set during cluster creation.

Step 2 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:

kinit *MRS cluster user*

Example: **kinit admin**

Step 4 Run the following commands to start Hive Beeline to create a table on Alluxio.

beeline

CREATE TABLE u_user (id int, name string, company string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LOCATION 'alluxio://<*Name of the Alluxio node*>:19998/u_user';

insert into u_user values(1,'Alice','Company A'),(2, 'Bob', 'Company B');

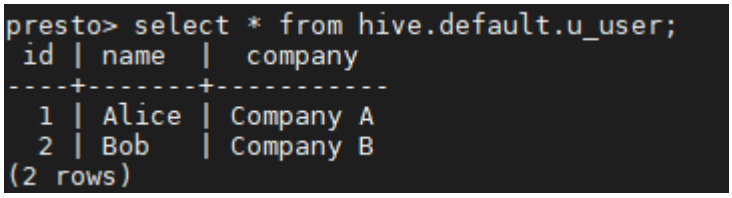
NOTE

Replace ***Name of the Alluxio node*>:19998** with the actual node name and port numbers of all nodes where the AlluxioMaster instance is deployed. Use commas (,) to separate the node name and port number, for example, **node-ana-coremspb.mrs-m0va.com:19998,node-master2kiww.mrs-m0va.com:19998,node-master1cqwv.mrs-m0va.com:19998**.

Step 5 Start the Presto client. For details, see **Step 2** to **Step 8** in **Using a Client to Execute Query Statements**.

Step 6 On the Presto client, run the **select * from hive.default.u_user;** statement to query the table created in Alluxio:

Figure 12-2 Using Presto to query the table created in Alluxio



----End

12.1.3 Common Operations of Alluxio

Preparations

1. Create a cluster with Alluxio installed.
2. Log in to the active Master node in a cluster as user **root** using the password set during cluster creation.
3. Run the following command to configure environment variables:

   source /opt/client/bigdata_env

Using the Alluxio Shell

The **Alluxio shell** contains multiple command line operations that interact with Alluxio.

- View a file system operation command list:

  alluxio fs

- Run the **ls** command to list the files in Alluxio. For example, list all files in the root directory:

  alluxio fs ls /

- Run the **copyFromLocal** command to copy local files to Alluxio:

  alluxio fs copyFromLocal /home/test_input.txt /test_input.txt

  Command output:

  ```
  Copied file:///home/test_input.txt to /test_input.txt
  ```

- Run the **ls** command again to list the files in Alluxio. The copied **test_input.txt** file is listed:

  alluxio fs ls /

  Command output:

  ```
  12      PERSISTED 11-28-2019 17:10:17:449 100% /test_input.txt
  ```

  The **test_input.txt** file is displayed in Alluxio. The parameters in the file indicate the file size, whether the file is persistent, creation date, cache ratio of the file in Alluxio, and file name.

- Run the **cat** command to print file content:

  alluxio fs cat /test_input.txt

  Command output:

  ```
  Test Alluxio
  ```

Mounting Function of Alluxio

Alluxio uses a unified namespace feature to unify the access to storage systems. For details, see **https://docs.alluxio.io/os/user/2.0/en/advanced/Namespace-Management.html**.

This feature allows users to mount different storage systems to an Alluxio namespace and seamlessly access files across storage systems through the Alluxio namespace.

1. Create a directory as a mount point in Alluxio.

   alluxio fs mkdir /mnt

   ```
   Successfully created directory /mnt
   ```

2. Mount an existing OBS file system to Alluxio. (Prerequisite: An agency with the **OBS OperateAccess** permission has been configured for the cluster. The **obs-mrstest** file system is used as an example. Replace the file system name with the actual one.

   alluxio fs mount /mnt/obs obs://obs-mrstest/data

   ```
   Mounted obs://obs-mrstest/data at /mnt/obs
   ```

3. List files in the OBS file system using the Alluxio namespace. Run the **ls** command to list the files in the OBS mount directory.

   alluxio fs ls /mnt/obs

   ```
   38      PERSISTED 11-28-2019 17:42:54:554   0% /mnt/obs/hive_load.txt
   12      PERSISTED 11-28-2019 17:43:07:743   0% /mnt/obs/test_input.txt
   ```

You can also view the newly mounted files and directories on the Alluxio web UI.

4. After the mounting is complete, you can seamlessly exchange data between different storage systems through the unified namespace of Alluxio. For example, run the **ls -R** command to list all files in a directory recursively:

   alluxio fs ls -R /

   ```
         0       PERSISTED 11-28-2019 11:15:19:719  DIR /app-logs
         1       PERSISTED 11-28-2019 11:18:36:885  DIR /apps
         1       PERSISTED 11-28-2019 11:18:40:209  DIR /apps/templeton
   239440292       PERSISTED 11-28-2019 11:18:40:209   0% /apps/templeton/hive.tar.gz
   .....
         1       PERSISTED 11-28-2019 19:00:23:879  DIR /mnt
         2       PERSISTED 11-28-2019 19:00:23:879  DIR /mnt/obs
        38       PERSISTED 11-28-2019 17:42:54:554   0% /mnt/obs/hive_load.txt
        12       PERSISTED 11-28-2019 17:43:07:743   0% /mnt/obs/test_input.txt
   .....
   ```

   The command output shows all files that are from the mounted storage system in the root directory of the Alluxio file system (the default directory is the HDFS root directory, that is, **hdfs://hacluster/**). The **/app-logs** and **/apps** directories are in HDFS, and the **/mnt/obs/** directory is in OBS.

Using Alluxio to Accelerate Data Access

Alluxio can accelerate data access, because it uses memory to store data. Example commands are provided as follows:

1. Upload the **test_data.csv** file (a sample that records recipes) to the **/data** directory of the **obs-mrstest** file system. Run the **ls** command to display the file status.

   alluxio fs ls /mnt/obs/test_data.csv

   ```
   294520189       PERSISTED 11-28-2019 19:38:55:000   0% /mnt/obs/test_data.csv
   ```

   The output indicates that the cache percentage of the file in Alluxio is 0%, that is, the file is not in Alluxio memory.

2. Count the occurrence times of the word "milk" in the file, and calculate the time consumed.

   time alluxio fs cat /mnt/obs/test_data.csv | grep -c milk

   ```
   52180

   real    0m10.765s
   user    0m5.540s
   sys     0m0.696s
   ```

3. Data is stored in memory after being read for the first time. When Alluxio reads data again, the data access speed is increased. For example, after running the **cat** command to obtain a file, run the **ls** command to check the file status.

   alluxio fs ls /mnt/obs/test_data.csv

   ```
   294520189       PERSISTED 11-28-2019 19:38:55:000 100% /mnt/obs/test_data.csv
   ```

   The output shows that the file has been fully loaded to Alluxio.

4. Access the file again, count the occurrence times of the word "eggs", and calculate the time consumed.

   time alluxio fs cat /mnt/obs/test_data.csv | grep -c eggs

   ```
   59510

   real    0m5.777s
   ```

```
user    0m5.992s
sys     0m0.592s
```

According to the comparison of the two time consumption records, the time consumed for accessing data stored in Alluxio memory is significantly reduced.

12.2 Using CarbonData (for Versions Earlier Than MRS 3.x)

12.2.1 Using CarbonData from Scratch

This section is for MRS 3.x or earlier. For MRS 3.x or later, see **Using CarbonData (for MRS 3.x or Later)**.

This section describes the procedure of using Spark CarbonData. All tasks are based on the Spark-beeline environment. The tasks include:

1. Connecting to Spark

   Before performing any operation on CarbonData, users must connect CarbonData to Spark.
2. Creating a CarbonData table

   After connecting to Spark, users must create a CarbonData table to load and query data.
3. Loading data to the CarbonData table

   Users load data from CSV files in HDFS to the CarbonData table.
4. Querying data from the CarbonData table

   After data is loaded to the CarbonData table, users can run query commands such as **groupby** and **where**.

Prerequisites

A client has been installed. For details, see **Using an MRS Client**.

Procedure

Step 1 Connect CarbonData to Spark.

1. Prepare a client based on service requirements and use user **root** to log in to the node where the client is installed.

   For example, if you have updated the client on the Master2 node, log in to the Master2 node to use the client. For details, see **Using an MRS Client**.
2. Run the following commands to switch the user and configure environment variables:

   sudo su - omm

   source /opt/client/bigdata_env
3. For clusters with Kerberos authentication enabled, run the following command to authenticate the user. For clusters with Kerberos authentication disabled, skip this step.

kinit *Spark username*

NOTE

The user needs to be added to user groups **hadoop** (primary group) and **hive**.

4. Run the following command to connect to the Spark environment.

   spark-beeline

Step 2 Create a CarbonData table.

Run the following command to create a CarbonData table, which is used to load and query data.

CREATE TABLE x1 (imei string, deviceInformationId int, mac string, productdate timestamp, updatetime timestamp, gamePointId double, contractNumber double)

STORED BY 'org.apache.carbondata.format'

TBLPROPERTIES ('DICTIONARY_EXCLUDE'='mac','DICTIONARY_INCLUDE'='deviceInformationId');

The command output is as follows:

```
+---------+--+
| result  |
+---------+--+
+---------+--+
No rows selected (1.551 seconds)
```

Step 3 Load data from CSV files to the CarbonData table.

Run the command to load data from CSV files based on the required parameters. Only CSV files are supported. The CSV column name and sequence configured in the **LOAD** command must be consistent with those in the CarbonData table. The data formats and number of data columns in the CSV files must also be the same as those in the CarbonData table.

The CSV files must be stored on HDFS. You can upload the files to OBS and import them from OBS to HDFS on the **Files** page of the MRS console.

If Kerberos authentication is enabled, prepare the CSV files in the work environment and import them to HDFS using open-source HDFS commands. In addition, assign the Spark user with the read and execute permissions of the files on HDFS by referring to **5**.

For example, the **data.csv** file is saved in the **tmp** directory of HDFS with the following contents:

```
x123,111,dd,2017-04-20 08:51:27,2017-04-20 07:56:51,2222,33333
```

The command for loading data from that file is as follows:

LOAD DATA inpath 'hdfs://hacluster/tmp/data.csv' into table x1 options('DELIMITER'=',','QUOTECHAR'='"','FILEHEADER'='imei,deviceinformationid,mac,productdate,updatetime,gamepointid,contractnumber');

The command output is as follows:

```
+---------+--+
| Result  |
+---------+--+
+---------+--+
No rows selected (3.039 seconds)
```

Step 4 Query data from the CarbonData.

- **Obtaining the number of records**

  Run the following command to obtain the number of records in the CarbonData table:

  select count(*) from x1;

- **Querying with the groupby condition**

  Run the following command to obtain the **deviceinformationid** records without repetition in the CarbonData table:

  select deviceinformationid,count (distinct deviceinformationid) from x1 group by deviceinformationid;

- **Querying with the where condition**

  Run the following command to obtain specific **deviceinformationid** records:

  select * from x1 where deviceinformationid='111';

Step 5 Run the following command to exit the Spark environment.

!quit

----End

12.2.2 About CarbonData Table

Description

CarbonData tables are similar to tables in the relational database management system (RDBMS). RDBMS tables consist of rows and columns to store data. CarbonData tables have fixed columns and also store structured data. In CarbonData, data is saved in entity files.

Supported Data Types

CarbonData tables support the following data types:

- Int
- String
- BigInt
- Decimal
- Double
- TimeStamp

Table 12-1 describes the details about each data type.

Table 12-1 CarbonData data types

Data Type	Description
Int	4-byte signed integer ranging from -2,147,483,648 to 2,147,483,647 **NOTE** If a non-dictionary column is of the **int** data type, it is internally stored as the **BigInt** type.
String	The maximum character string length is 100000.
BigInt	Data is saved using the 64-bit technology. The value ranges from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.
Decimal	The default value is (10,0) and maximum value is (38,38). **NOTE** When query with filters, append **BD** to the number to achieve accurate results. For example, **select * from carbon_table where num = 1234567890123456.22BD**.
Double	Data is saved using the 64-bit technology. The value ranges from 4.9E-324 to 1.7976931348623157E308.
TimeStamp	**yyyy-MM-dd HH:mm:ss** format is used by default.

NOTE

Measurement of all Integer data is processed and displayed using the **BigInt** data type.

12.2.3 Creating a CarbonData Table

Scenario

A CarbonData table must be created to load and query data.

Creating a Table with Self-Defined Columns

Users can create a table by specifying its columns and data types. For analysis clusters with Kerberos authentication enabled, if a user wants to create a CarbonData table in a database other than the **default** database, the **Create** permission of the database must be added to the role to which the user is bound in Hive role management.

Sample command:

CREATE TABLE IF NOT EXISTS productdb.productSalesTable (

productNumber Int,

productName String,

storeCity String,

storeProvince String,

revenue Int)

STORED BY 'org.apache.carbondata.format'

TBLPROPERTIES (

'table_blocksize'='128',

'DICTIONARY_EXCLUDE'='productName',

'DICTIONARY_INCLUDE'='productNumber');

The following table describes parameters of preceding commands.

Table 12-2 Parameter description

Parameter	Description
productSalesTable	Table name. The table is used to load data for analysis. The table name consists of letters, digits, and underscores (_).
productdb	Database name. The database maintains logical connections with tables stored in it to identify and manage the tables. The database name consists of letters, digits, and underscores (_).
productNumber productName storeCity storeProvince revenue	Columns in the table. The columns are service entities for data analysis. The column name (field name) consists of letters, digits, and underscores (_). **NOTE** In CarbonData, you cannot configure a column's NOT NULL or default value, or the primary key of the table.
table_blocksize	Block size of data files used by the CarbonData table. The value ranges from 1 MB to 2048 MB. The default is 1024 MB. ● If the value of **table_blocksize** is too small, a large number of small files will be generated when data is loaded. This may affect the performance in using HDFS. ● If the value of **table_blocksize** is too large, a large volume of data must be read from a block and the read concurrency is low when data is queried. As a result, the query performance deteriorates. You are advised to set the block size based on the data volume. For example, set the block size to 256 MB for GB-level data, 512 MB for TB-level data, and 1024 MB for PB-level data.

Parameter	Description
DICTIONARY_EXCLUDE	Specifies the columns that do not generate dictionaries. This function is optional and applicable to columns of high complexity. By default, the system generates dictionaries for columns of the String type. However, as the number of values in the dictionaries increases, conversion operations by the dictionaries increase and the system performance deteriorates. Generally, if a column has over 50,000 unique data records, it is considered as a highly complex column and dictionary generation must be disabled. **NOTE** Non-dictionary columns support only the String and Timestamp data types.
DICTIONARY_INCLUDE	Specifies the columns that generate dictionaries. This function is optional and applicable to columns of low complexity. It improves the performance of queries with the **groupby** condition. Generally, the complexity of a dictionary column cannot exceed 50,000.

12.2.4 Deleting a CarbonData Table

Scenario

Unused CarbonData tables can be deleted. After a CarbonData table is deleted, its metadata and loaded data are deleted together.

Procedure

Step 1 Run the following command to delete a CarbonData table:

DROP TABLE *[IF EXISTS] [db_name.]table_name*;

db_name is optional. If **db_name** is not specified, the table named **table_name** in the current database is deleted.

For example, run the following command to delete the **productSalesTable** table in the **productdb** database:

DROP TABLE productdb.productSalesTable;

Step 2 Run the following command to confirm that the table is deleted:

SHOW TABLES;

----End

12.3 Using CarbonData (for MRS 3.x or Later)

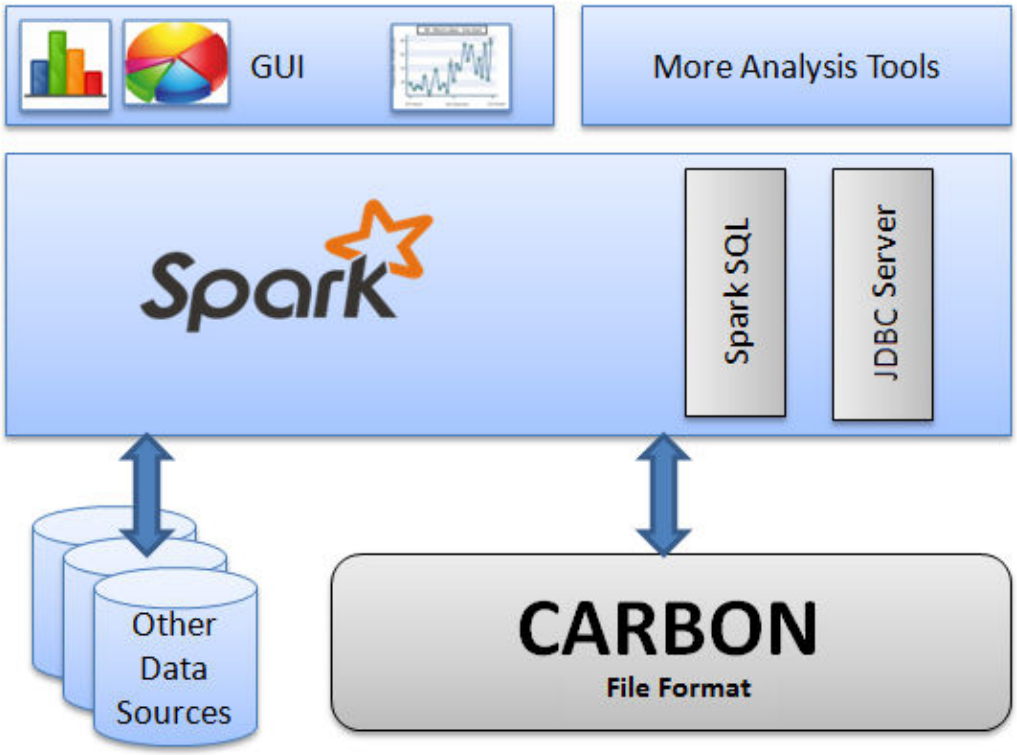
12.3.1 Overview

This section is for MRS 3.x or later. For MRS 3.x or earlier, see **Using CarbonData (for Versions Earlier Than MRS 3.x)**.

12.3.1.1 CarbonData Overview

CarbonData is a new Apache Hadoop native data-store format. CarbonData allows faster interactive queries over PetaBytes of data using advanced columnar storage, index, compression, and encoding techniques to improve computing efficiency. In addition, CarbonData is also a high-performance analysis engine that integrates data sources with Spark.

Figure 12-3 Basic architecture of CarbonData



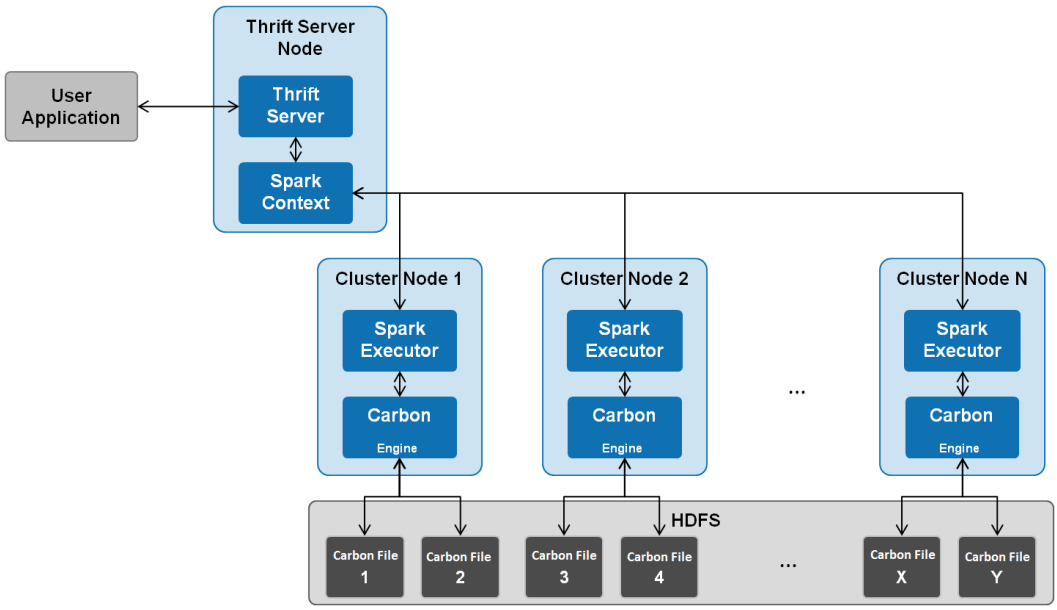
The purpose of using CarbonData is to provide quick response to ad hoc queries of big data. Essentially, CarbonData is an Online Analytical Processing (OLAP) engine, which stores data by using tables similar to those in Relational Database Management System (RDBMS). You can import more than 10 TB data to tables created in CarbonData format, and CarbonData automatically organizes and stores data using the compressed multi-dimensional indexes. After data is loaded to CarbonData, CarbonData responds to ad hoc queries in seconds.

CarbonData integrates data sources into the Spark ecosystem and you can query and analyze the data using Spark SQL. You can also use the third-party tool JDBCServer provided by Spark to connect to SparkSQL.

Topology of CarbonData

CarbonData runs as a data source inside Spark. Therefore, CarbonData does not start any additional processes on nodes in clusters. CarbonData engine runs inside the Spark executor.

Figure 12-4 Topology of CarbonData



Data stored in CarbonData Table is divided into several CarbonData data files. Each time when data is queried, CarbonData Engine reads and filters data sets. CarbonData Engine runs as a part of the Spark Executor process and is responsible for handling a subset of data file blocks.

Table data is stored in HDFS. Nodes in the same Spark cluster can be used as HDFS data nodes.

CarbonData Features

- SQL: CarbonData is compatible with Spark SQL and supports SQL query operations performed on Spark SQL.
- Simple Table dataset definition: CarbonData allows you to define and create datasets by using user-friendly Data Definition Language (DDL) statements. CarbonData DDL is flexible and easy to use, and can define complex tables.
- Easy data management: CarbonData provides various data management functions for data loading and maintenance. CarbonData supports bulk loading of historical data and incremental loading of new data. Loaded data can be deleted based on load time and a specific loading operation can be undone.
- CarbonData file format is a columnar store in HDFS. This format has many new column-based file storage features, such as table splitting and data compression. CarbonData has the following characteristics:
  - Stores data along with index: Significantly accelerates query performance and reduces the I/O scans and CPU resources, when there are filters in the query. CarbonData index consists of multiple levels of indices. A

processing framework can leverage this index to reduce the task that needs to be schedules and processed, and it can also perform skip scan in more finer grain unit (called blocklet) in task side scanning instead of scanning the whole file.

- Operable encoded data: Through supporting efficient compression, CarbonData can query on compressed/encoded data. The data can be converted just before returning the results to the users, which is called late materialized.
- Support for various use cases with one single data format: like interactive OLAP-style query, sequential access (big scan), and random access (narrow scan).

Key Technologies and Advantages of CarbonData

- Quick query response: CarbonData features high-performance query. The query speed of CarbonData is 10 times of that of Spark SQL. It uses dedicated data formats and applies multiple index technologies and multiple push-down optimizations, providing quick response to TB-level data queries.
- Efficient data compression: CarbonData compresses data by combining the lightweight and heavyweight compression algorithms. This significantly saves 60% to 80% data storage space and the hardware storage cost.

12.3.1.2 Main Specifications of CarbonData

Main Specifications of CarbonData

Table 12-3 Main Specifications of CarbonData

Entity	Tested Value	Test Environment
Number of tables	10000	3 nodes. 4 vCPUs and 20 GB memory for each executor. Driver memory: 5 GB, 3 executors. Total columns: 107 String: 75 Int: 13 BigInt: 7 Timestamp: 6 Double: 6
Number of table columns	2000	3 nodes. 4 vCPUs and 20 GB memory for each executor. Driver memory: 5 GB, 3 executors.
Maximum size of a raw CSV file	200 GB	17 cluster nodes. 150 GB memory and 25 vCPUs for each executor. Driver memory: 10 GB, 17 executors.

Entity	Tested Value	Test Environment
Number of CSV files in each folder	100 folders. Each folder has 10 files. The size of each file is 50 MB.	3 nodes. 4 vCPUs and 20 GB memory for each executor. Driver memory: 5 GB, 3 executors.
Number of load folders	10000	3 nodes. 4 vCPUs and 20 GB memory for each executor. Driver memory: 5 GB, 3 executors.

The memory required for data loading depends on the following factors:

- Number of columns
- Column values
- Concurrency (configured using **carbon.number.of.cores.while.loading**)
- Sort size in memory (configured using **carbon.sort.size**)
- Intermediate cache (configured using **carbon.graph.rowset.size**)

Data loading of an 8 GB CSV file that contains 10 million records and 300 columns with each row size being about 0.8 KB requires about 10 GB executor memory. That is, set **carbon.sort.size** to **100000** and retain the default values for other parameters.

Table Specifications

Table 12-4 Table specifications

Entity	Tested Value
Number of secondary index tables	10
Number of composite columns in a secondary index table	5
Length of column name in a secondary index table (unit: character)	120
Length of a secondary index table name (unit: character)	120
Cumulative length of all secondary index table names + column names in an index table* (unit: character)	3800**

NOTE

- * Characters of column names in an index table refer to the upper limit allowed by Hive or the upper limit of available resources.
- ** Secondary index tables are registered using Hive and stored in HiveSERDEPROPERTIES in JSON format. The value of **SERDEPROPERTIES** supported by Hive can contain a maximum of 4,000 characters and cannot be changed.

12.3.2 Configuration Reference

This section provides the details of all the configurations required for the CarbonData System.

Table 12-5 System configurations in **carbon.properties**

Parameter	Default Value	Description
carbon.ddl.base.hdfs.url	hdfs://hacluster/opt/data	HDFS relative path from the HDFS base path, which is configured in **fs.defaultFS**. The path configured in **carbon.ddl.base.hdfs.url** will be appended to the HDFS path configured in **fs.defaultFS**. If this path is configured, you do not need to pass the complete path while dataload. For example, if the absolute path of the CSV file is **hdfs://10.18.101.155:54310/data/cnbc/2016/xyz.csv**, the path **hdfs://10.18.101.155:54310** will come from property **fs.defaultFS** and you can configure **/data/cnbc/** as **carbon.ddl.base.hdfs.url**. During data loading, you can specify the CSV path as **/2016/xyz.csv**.
carbon.badRecords.location	-	Storage path of bad records. This path is an HDFS path. The default value is **Null**. If bad records logging or bad records operation redirection is enabled, the path must be configured by the user.
carbon.bad.records.action	fail	The following are four types of actions for bad records: **FORCE**: Data is automatically corrected by storing the bad records as NULL. **REDIRECT**: Bad records are written to the raw CSV instead of being loaded. **IGNORE**: Bad records are neither loaded nor written to the raw CSV. **FAIL**: Data loading fails if any bad records are found.

Parameter	Default Value	Description
carbon.update.sync.folder	/tmp/carbondata	Specifies the **modifiedTime.mdt** file path. You can set it to an existing path or a new path. **NOTE** If you set this parameter to an existing path, ensure that all users can access the path and the path has the 777 permission.

Table 12-6 Performance configurations in **carbon.properties**

Parameter	Default Value	Description
Data Loading Configuration		
carbon.sort.file.write.buffer.size	16384	CarbonData sorts data and writes it to a temporary file to limit memory usage. This parameter controls the size of the buffer used for reading and writing temporary files. The unit is bytes. The value ranges from 10240 to 10485760.
carbon.graph.rowset.size	100,000	Rowset size exchanged in data loading graph steps. The value ranges from 500 to 1,000,000.
carbon.number.of.cores.while.loading	6	Number of cores used during data loading. The greater the number of cores, the better the compaction performance. If the CPU resources are sufficient, you can increase the value of this parameter.
carbon.sort.size	500000	Number of records to be sorted
carbon.enableXXHash	true	Hashmap algorithm used for hashkey calculation
carbon.number.of.cores.block.sort	7	Number of cores used for sorting blocks during data loading
carbon.max.driver.lru.cache.size	-1	Maximum size of LRU caching for data loading at the driver side. The unit is MB. The default value is **-1**, indicating that there is no memory limit for the caching. Only integer values greater than 0 are accepted.
carbon.max.executor.lru.cache.size	-1	Maximum size of LRU caching for data loading at the executor side. The unit is MB. The default value is **-1**, indicating that there is no memory limit for the caching. Only integer values greater than 0 are accepted. If this parameter is not configured, the value of **carbon.max.driver.lru.cache.size** is used.

Parameter	Default Value	Description
carbon.merge.sort.prefetch	true	Whether to enable prefetch of data during merge sort while reading data from sorted temp files in the process of data loading
carbon.update.persist.enable	true	Configuration to enable the dataset of RDD/dataframe to persist data. Enabling this will reduce the execution time of UPDATE operation.
enable.unsafe.sort	true	Whether to use unsafe sort during data loading. Unsafe sort reduces the garbage collection during data load operation, resulting in better performance. The default value is **true**, indicating that unsafe sort is enabled.
enable.offheap.sort	true	Whether to use off-heap memory for sorting of data during data loading
offheap.sort.chunk.size.inmb	64	Size of data chunks to be sorted, in MB. The value ranges from 1 to 1024.
carbon.unsafe.working.memory.in.mb	512	Size of the unsafe working memory. This will be used for sorting data and storing column pages. The unit is MB. Memory required for data loading: carbon.number.of.cores.while.loading [default value is 6] x Number of tables to load in parallel x offheap.sort.chunk.size.inmb [default value is 64 MB] + carbon.blockletgroup.size.in.mb [default value is 64 MB] + Current compaction ratio [64 MB/3.5]) = Around 900 MB per table Memory required for data query: (SPARK_EXECUTOR_INSTANCES. [default value is 2] x (carbon.blockletgroup.size.in.mb [default value: 64 MB] + carbon.blockletgroup.size.in.mb [default value = 64 MB x 3.5) x Number of cores per executor [default value: 1]) = ~ 600 MB
carbon.sort.inmemory.storage.size.in.mb	512	Size of the intermediate sort data to be kept in the memory. Once the specified value is reached, the system writes data to the disk. The unit is MB.

Parameter	Default Value	Description
sort.inmemory.size.inmb	1024	Size of the intermediate sort data to be kept in the memory. Once the specified value is reached, the system writes data to the disk. The unit is MB. If **carbon.unsafe.working.memory.in.mb** and **carbon.sort.inmemory.storage.size.in.mb** are configured, you do not need to set this parameter. If this parameter has been configured, 20% of the memory is used for working memory **carbon.unsafe.working.memory.in.mb**, and 80% is used for sort storage memory **carbon.sort.inmemory.storage.size.in.mb**. **NOTE** The value of **spark.yarn.executor.memoryOverhead** configured for Spark must be greater than the value of **sort.inmemory.size.inmb** configured for CarbonData. Otherwise, Yarn might stop the executor if off-heap access exceeds the configured executor memory.
carbon.blockletgroup.size.in.mb	64	The data is read as a group of blocklets which are called blocklet groups. This parameter specifies the size of each blocklet group. Higher value results in better sequential I/O access. The minimum value is 16 MB. Any value less than 16 MB will be reset to the default value (64 MB). The unit is MB.
enable.inmemory.merge.sort	false	Whether to enable **inmemorymerge sort**.
use.offheap.in.query.processing	true	Whether to enable **offheap** in query processing.
carbon.load.sort.scope	local_sort	Sort scope for the load operation. There are two types of sort: **batch_sort** and **local_sort**. If **batch_sort** is selected, the loading performance is improved but the query performance is reduced.
carbon.batch.sort.size.inmb	-	Size of data to be considered for batch sorting during data loading. The recommended value is less than 45% of the total sort data. The unit is MB. **NOTE** If this parameter is not set, its value is about 45% of the value of **sort.inmemory.size.inmb** by default.
enable.unsafe.columnpage	true	Whether to keep page data in heap memory during data loading or query to prevent garbage collection bottleneck.

Parameter	Default Value	Description
carbon.use.local.dir	false	Whether to use Yarn local directories for multi-disk data loading. If this parameter is set to **true**, Yarn local directories are used to load multi-disk data to improve data loading performance.
carbon.use.multiple.temp.dir	false	Whether to use multiple temporary directories for storing temporary files to improve data loading performance.
carbon.load.datamaps.parallel.db_name.table_name	N/A	The value can be **true** or **false**. You can set the database name and table name to improve the first query performance of the table.
Compaction Configuration		
carbon.number.of.cores.while.compacting	2	Number of cores to be used while compacting data. The greater the number of cores, the better the compaction performance. If the CPU resources are sufficient, you can increase the value of this parameter.
carbon.compaction.level.threshold	4,3	This configuration is for minor compaction which decides how many segments to be merged. For example, if this parameter is set to **2,3**, minor compaction is triggered every two segments. **3** is the number of level 1 compacted segments which is further compacted to new segment. The value ranges from 0 to 100.
carbon.major.compaction.size	1024	Major compaction size. Sum of the segments which is below this threshold will be merged. The unit is MB.
carbon.horizontal.compaction.enable	true	Whether to enable/disable horizontal compaction. After every DELETE and UPDATE statement, horizontal compaction may occur in case the incremental (DELETE/ UPDATE) files becomes more than specified threshold. By default, this parameter is set to **true**. You can set this parameter to **false** to disable horizontal compaction.
carbon.horizontal.update.compaction.threshold	1	Threshold limit on number of UPDATE delta files within a segment. In case the number of delta files goes beyond the threshold, the UPDATE delta files within the segment becomes eligible for horizontal compaction and are compacted into single UPDATE delta file. By default, this parameter is set to **1**. The value ranges from **1** to **10000**.

Parameter	Default Value	Description
carbon.horizontal.delete.compaction.threshold	1	Threshold limit on number of DELETE incremental files within a block of a segment. In case the number of incremental files goes beyond the threshold, the DELETE incremental files for the particular block of the segment becomes eligible for horizontal compaction and are compacted into single DELETE incremental file. By default, this parameter is set to **1**. The value ranges from **1** to **10000**.
Query Configuration		
carbon.number.of.cores	4	Number of cores to be used during query
carbon.limit.block.distribution.enable	false	Whether to enable the CarbonData distribution for limit query. The default value is **false**, indicating that block distribution is disabled for query statements that contain the keyword limit. For details about how to optimize this parameter, see **Configurations for Performance Tuning**.
carbon.custom.block.distribution	false	Whether to enable Spark or CarbonData block distribution. By default, the value is **false**, indicating that Spark block distribution is enabled. To enable CarbonData block distribution, change the value to **true**.
carbon.infilter.subquery.pushdown.enable	false	If this is set to **true** and a Select query is triggered in the filter with subquery, the subquery is executed and the output is broadcast as IN filter to the left table. Otherwise, SortMergeSemiJoin is executed. You are advised to set this to **true** when IN filter subquery does not return too many records. For example, when the IN sub-sentence query returns 10,000 or fewer records, enabling this parameter will give the query results faster. Example: ***select * from flow_carbon_256b where cus_no in (select cus_no from flow_carbon_256b where dt>='20260101' and dt<='20260701' and txn_bk='tk_1' and txn_br='tr_1') limit 1000;***
carbon.scheduler.minRegisteredResourcesRatio	0.8	Minimum resource (executor) ratio needed for starting the block distribution. The default value is **0.8**, indicating that 80% of the requested resources are allocated for starting block distribution.
carbon.dynamicAllocation.schedulerTimeout	5	Maximum time that the scheduler waits for executors to be active. The default value is **5** seconds, and the maximum value is **15** seconds.

Parameter	Default Value	Description
enable.unsafe.in.query.processing	true	Whether to use unsafe sort during query. Unsafe sort reduces the garbage collection during query, resulting in better performance. The default value is **true**, indicating that unsafe sort is enabled.
carbon.enable.vector.reader	true	Whether to enable vector processing for result collection to improve query performance
carbon.query.show.datamaps	true	**SHOW TABLES** lists all tables including the primary table and datamaps. To filter out the datamaps, set this parameter to **false**.
Secondary Index Configuration		
carbon.secondary.index.creation.threads	1	Number of threads to concurrently process segments during secondary index creation. This property helps fine-tuning the system when there are a lot of segments in a table. The value ranges from 1 to 50.
carbon.si.lookup.partialstring	true	• When the parameter value is **true**, it includes indexes started with, ended with, and contained. • When the parameter value is **false**, it includes only secondary indexes started with.
carbon.si.segment.merge	true	Enabling this property merges **.carbondata** files inside the secondary index segment. The merging will happen after the load operation. That is, at the end of the secondary index table load, small files are checked and merged. **NOTE** Table Block Size is used as the size threshold for merging small files.

Table 12-7 Other configurations in **carbon.properties**

Parameter	Default Value	Description
Data Loading Configuration		

Parameter	Default Value	Description
carbon.lock.type	HDFSLOCK	Type of lock to be acquired during concurrent operations on a table. There are following types of lock implementation: • **LOCALLOCK**: Lock is created on local file system as a file. This lock is useful when only one Spark driver (or JDBCServer) runs on a machine. • **HDFSLOCK**: Lock is created on HDFS file system as a file. This lock is useful when multiple Spark applications are running and no ZooKeeper is running on a cluster.
carbon.sort.intermediate.files.limit	20	Minimum number of intermediate files. After intermediate files are generated, sort and merge the files. For details about how to optimize this parameter, see **Configurations for Performance Tuning**.
carbon.csv.read.buffersize.byte	1048576	Size of CSV reading buffer
carbon.merge.sort.reader.thread	3	Maximum number of threads used for reading intermediate files for final merging.
carbon.concurrent.lock.retries	100	Maximum number of retries used to obtain the concurrent operation lock. This parameter is used for concurrent loading.
carbon.concurrent.lock.retry.timeout.sec	1	Interval between the retries to obtain the lock for concurrent operations.
carbon.lock.retries	3	Maximum number of retries to obtain the lock for any operations other than import.
carbon.lock.retry.timeout.sec	5	Interval between the retries to obtain the lock for any operation other than import.
carbon.tempstore.location	/opt/Carbon/TempStoreLoc	Temporary storage location. By default, the **System.getProperty("java.io.tmpdir")** method is used to obtain the value. For details about how to optimize this parameter, see the description of **carbon.use.local.dir** in **Configurations for Performance Tuning**.
carbon.load.log.counter	500000	Data loading records count in logs

Parameter	Default Value	Description
SERIALIZATION_NULL_FORMAT	\N	Value to be replaced with NULL
carbon.skip.empty.line	false	Setting this property will ignore the empty lines in the CSV file during data loading.
carbon.load.datamaps.parallel	false	Whether to enable parallel datamap loading for all tables in all sessions. This property will improve the time to load datamaps into memory by distributing the job among executors, thus improving query performance.
Merging Configuration		
carbon.numberof.preserve.segments	0	If you want to preserve some number of segments from being compacted, then you can set this configuration. For example, if **carbon.numberof.preserve.segments** is set to **2**, the latest two segments will always be excluded from the compaction. No segments will be preserved by default.
carbon.allowed.compaction.days	0	This configuration is used to control on the number of recent segments that needs to be merged. For example, if this parameter is set to **2**, the segments which are loaded in the time frame of past 2 days only will get merged. Segments which are loaded earlier than 2 days will not be merged. This configuration is disabled by default.
carbon.enable.auto.load.merge	false	Whether to enable compaction along with data loading.
carbon.merge.index.in.segment	true	This configuration enables to merge all the CarbonIndex files (**.carbonindex**) into a single MergeIndex file (**.carbonindexmerge**) upon data loading completion. This significantly reduces the delay in serving the first query.
Query Configuration		
max.query.execution.time	60	Maximum time allowed for one query to be executed. The unit is minute.

Parameter	Default Value	Description
carbon.enableMinMax	true	MinMax is used to improve query performance. You can set this to **false** to disable this function.
carbon.lease.recovery.retry.count	5	Maximum number of attempts that need to be made for recovering a lease on a file. Minimum value: **1** Maximum value: **50**
carbon.lease.recovery.retry.interval	1000 (ms)	Interval or pause time after a lease recovery attempt is made on a file. Minimum value: **1000** (ms) Maximum value: **10000** (ms)

Table 12-8 Spark configuration reference in **spark-defaults.conf**

Parameter	Default Value	Description
spark.driver.memory	4G	Memory to be used for the driver process. SparkContext has been initialized. **NOTE** In client mode, do not use SparkConf to set this parameter in the application because the driver JVM has been started. To configure this parameter, configure it in the **--driver-memory** command-line option or in the default property file.
spark.executor.memory	4 GB	Memory to be used for each executor process.
spark.sql.crossJoin.enabled	true	If the query contains a cross join, enable this property so that no error is thrown. In this case, you can use a cross join instead of a join for better performance.

Configure the following parameters in the **spark-defaults.conf** file on the Spark driver.

- In spark-sql mode:

Table 12-9 Parameter description

Parameter	Value	Description
spark.driver.extraJavaOptions	-Dlog4j.configuration=file:/opt/client/Spark2x/spark/conf/log4j.properties -Djetty.version=x.y.z -Dzookeeper.server.principal=zookeeper/hadoop.*<System domain name>* -Djava.security.krb5.conf=/opt/client/KrbClient/kerberos/var/krb5kdc/krb5.conf -Djava.security.auth.login.config=/opt/client/Spark2x/spark/conf/jaas.conf -Dorg.xerial.snappy.tempdir=/opt/client/Spark2x/tmp -Dcarbon.properties.filepath=/opt/client/Spark2x/spark/conf/carbon.properties -Djava.io.tmpdir=/opt/client/Spark2x/tmp	The default value **/opt/client/Spark2x/spark** indicates **CLIENT_HOME** of the client and is added to the end of the value of **spark.driver.extraJavaOptions**. This parameter is used to specify the path of the **carbon.properties**file in Driver. **NOTE** Spaces next to equal marks (=) are not allowed.
spark.sql.session.state.builder	org.apache.spark.sql.hive.FIHiveACLSessionStateBuilder	Session state constructor.
spark.carbon.sqlastbuilder.classname	org.apache.spark.sql.hive.CarbonInternalSqlAstBuilder	AST constructor.
spark.sql.catalog.class	org.apache.spark.sql.hive.HiveACLExternalCatalog	Hive External catalog to be used. This parameter is mandatory if Spark ACL is enabled.
spark.sql.hive.implementation	org.apache.spark.sql.hive.HiveACLClientImpl	How to call the Hive client. This parameter is mandatory if Spark ACL is enabled.
spark.sql.hiveClient.isolation.enabled	false	This parameter is mandatory if Spark ACL is enabled.

- In JDBCServer mode:

Table 12-10 Parameter description

Parameter	Value	Description
spark.driver.extraJavaOptions	-Xloggc:${SPARK_LOG_DIR}/indexserver-omm-%p-gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:MaxDirectMemorySize=512M -XX:MaxMetaspaceSize=512M -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=10M -XX:OnOutOfMemoryError='kill -9 %p' -Djetty.version=x.y.z -Dorg.xerial.snappy.tempdir=${BIGDATA_HOME}/tmp/spark2x/JDBCServer/snappy_tmp -Djava.io.tmpdir=${BIGDATA_HOME}/tmp/spark2x/JDBCServer/io_tmp -Dcarbon.properties.filepath=${SPARK_CONF_DIR}/carbon.properties -Djdk.tls.ephemeralDHKeySize=20	The default value **${SPARK_CONF_DIR}** depends on a specific cluster and is added to the end of the value of the **spark.driver.extraJavaOptions** parameter. This parameter is used to specify the path of the **carbon.properties** file in Driver. **NOTE** Spaces next to equal marks (=) are not allowed.

Parameter	Value	Description
	48 -Dspark.ssl.keyStore=${SPARK_CONF_DIR}/child.keystore #{java_stack_prefer}	
spark.sql.session.state.builder	org.apache.spark.sql.hive.FIHiveACLSessionStateBuilder	Session state constructor.
spark.carbon.sqlastbuilder.classname	org.apache.spark.sql.hive.CarbonInternalSqlAstBuilder	AST constructor.
spark.sql.catalog.class	org.apache.spark.sql.hive.HiveACLExternalCatalog	Hive External catalog to be used. This parameter is mandatory if Spark ACL is enabled.
spark.sql.hive.implementation	org.apache.spark.sql.hive.HiveACLClientImpl	How to call the Hive client. This parameter is mandatory if Spark ACL is enabled.
spark.sql.hiveClient.isolation.enabled	false	This parameter is mandatory if Spark ACL is enabled.

12.3.3 CarbonData Operation Guide

12.3.3.1 CarbonData Quick Start

This section describes how to create CarbonData tables, load data, and query data. This quick start provides operations based on the Spark Beeline client. If you want to use Spark shell, wrap the queries with **spark.sql()**.

The following describes how to load data from a CSV file to a CarbonData table.

Table 12-11 CarbonData Quick Start

Operation	Description
Preparing a CSV File	Prepare the CSV file to be loaded to the CarbonData Table.

Operation	Description
Connecting to CarbonData	Connect to CarbonData before performing any operations on CarbonData.
Creating a CarbonData Table	Create a CarbonData table to load data and perform query operations.
Loading Data to a CarbonData Table	Load data from CSV to the created table.
Querying Data from a CarbonData Table	Perform query operations such as filters and groupby.

Preparing a CSV File

1. Prepare a CSV file named **test.csv** on the local PC. An example is as follows:

   ```
   13418592122,1001, MAC address, 2017-10-23 15:32:30,2017-10-24 15:32:30,62.50,74.56
   13418592123 1002,  MAC address, 2017-10-23 16:32:30,2017-10-24 16:32:30,17.80,76.28
   13418592124,1003, MAC address, 2017-10-23 17:32:30,2017-10-24 17:32:30,20.40,92.94
   13418592125 1004,  MAC address, 2017-10-23 18:32:30,2017-10-24 18:32:30,73.84,8.58
   13418592126,1005, MAC address, 2017-10-23 19:32:30,2017-10-24 19:32:30,80.50,88.02
   13418592127 1006,  MAC address, 2017-10-23 20:32:30,2017-10-24 20:32:30,65.77,71.24
   13418592128,1007, MAC address, 2017-10-23 21:32:30,2017-10-24 21:32:30,75.21,76.04
   13418592129,1008, MAC address, 2017-10-23 22:32:30,2017-10-24 22:32:30,63.30,94.40
   13418592130, 1009, MAC address, 2017-10-23 23:32:30,2017-10-24 23:32:30,95.51,50.17
   13418592131,1010, MAC address, 2017-10-24 00:32:30,2017-10-25 00:32:30,39.62,99.13
   ```

2. Use WinSCP to import the CSV file to the directory of the node where the client is installed, for example, **/opt**.
3. Log in to FusionInsight Manager and choose **System**. In the navigation pane on the left, choose **Permission** > **User**, click **Create** to create human-machine user **sparkuser**, and add the user to user groups hadoop (primary group) and hive.
4. Run the following commands to go to the client installation directory, load environment variables, and authenticate the user.

   cd /*Client installation directory*

   source ./bigdata_env

   source ./Spark2x/component_env

   kinit sparkuser
5. Run the following command to upload the CSV file to the **/data** directory of the HDFS.

   hdfs dfs -put /opt/test.csv /data/

Connecting to CarbonData

- Use Spark SQL or Spark shell to connect to Spark and run Spark SQL commands.
- Run the following commands to start the JDBCServer and use a JDBC client (for example, Spark Beeline) to connect to the JDBCServer.

  cd ./Spark2x/spark/bin

  ./spark-beeline

Creating a CarbonData Table

After connecting Spark Beeline with the JDBCServer, create a CarbonData table to load data and perform query operations. Run the following commands to create a simple table:

create table x1 (imei string, deviceInformationId int, mac string, productdate timestamp, updatetime timestamp, gamePointId double, contractNumber double) STORED AS carbondata TBLPROPERTIES ('SORT_COLUMNS'='imei,mac');

The command output is as follows:

```
+---------+
| Result  |
+---------+
+---------+
No rows selected (1.093 seconds)
```

Loading Data to a CarbonData Table

After you have created a CarbonData table, you can load the data from CSV to the created table.

Run the following command with required parameters to load data from CSV. The column names of the CarbonData table must match the column names of the CSV file.

LOAD DATA inpath 'hdfs://hacluster/data/*test.csv***' into table** ***x1*** **options('DELIMITER'=',', 'QUOTECHAR'='"','FILEHEADER'='imei, deviceinformationid,mac, productdate,updatetime, gamepointid,contractnumber');**

test.csv is the CSV file prepared in **Preparing a CSV File** and **x1** is the table name.

The CSV example file is as follows:

```
13418592122,1001, MAC address, 2017-10-23 15:32:30,2017-10-24 15:32:30,62.50,74.56
13418592123 1002,  MAC address, 2017-10-23 16:32:30,2017-10-24 16:32:30,17.80,76.28
13418592124,1003, MAC address, 2017-10-23 17:32:30,2017-10-24 17:32:30,20.40,92.94
13418592125 1004,  MAC address, 2017-10-23 18:32:30,2017-10-24 18:32:30,73.84,8.58
13418592126,1005, MAC address, 2017-10-23 19:32:30,2017-10-24 19:32:30,80.50,88.02
13418592127 1006,  MAC address, 2017-10-23 20:32:30,2017-10-24 20:32:30,65.77,71.24
13418592128,1007, MAC address, 2017-10-23 21:32:30,2017-10-24 21:32:30,75.21,76.04
13418592129,1008, MAC address, 2017-10-23 22:32:30,2017-10-24 22:32:30,63.30,94.40
13418592130, 1009, MAC address, 2017-10-23 23:32:30,2017-10-24 23:32:30,95.51,50.17
13418592131,1010, MAC address, 2017-10-24 00:32:30,2017-10-25 00:32:30,39.62,99.13
```

The command output is as follows:

```
+-------------+
|Segment ID  |
+-------------+
|0           |
+-------------+
No rows selected (3.039 seconds)
```

Querying Data from a CarbonData Table

After a CarbonData table is created and the data is loaded, you can perform query operations as required. Some query operations are provided as examples.

- **Obtaining the number of records**

  Run the following command to obtain the number of records in the CarbonData table:

  select count(*) from x1;

- **Querying with the groupby condition**

  Run the following command to obtain the **deviceinformationid** records without repetition in the CarbonData table:

  select deviceinformationid,count (distinct deviceinformationid) from x1 group by deviceinformationid;

- **Querying with Filter**

  Run the following command to obtain specific **deviceinformationid** records:

  select * from x1 where deviceinformationid='1010';

Using CarbonData on Spark-shell

If you need to use CarbonData on a Spark-shell, you need to create a CarbonData table, load data to the CarbonData table, and query data in CarbonData as follows:

```
spark.sql("CREATE TABLE x2(imei string, deviceInformationId int, mac string, productdate timestamp, updatetime timestamp, gamePointId double, contractNumber double) STORED AS carbondata")
spark.sql("LOAD DATA inpath 'hdfs://hacluster/data/x1_without_header.csv' into table x2 options('DELIMITER'=',', 'QUOTECHAR'='\"','FILEHEADER'='imei, deviceinformationid,mac, productdate,updatetime, gamepointid,contractnumber')")
spark.sql("SELECT * FROM x2").show()
```

12.3.3.2 CarbonData Table Management

12.3.3.2.1 About CarbonData Table

Overview

In CarbonData, data is stored in entities called tables. CarbonData tables are similar to RDBMS tables. RDBMS data is stored in a table consisting of rows and columns. CarbonData tables store structured data, and have fixed columns and data types.

Supported Data Types

CarbonData tables support the following data types:

- Int
- String
- BigInt
- Smallint
- Char
- Varchar
- Boolean
- Decimal

- Double
- TimeStamp
- Date
- Array
- Struct
- Map

The following table describes supported data types and their respective values range.

Table 12-12 CarbonData data types

Data Type	Value Range
Int	4-byte signed integer ranging from -2,147,483,648 to 2,147,483,647. **NOTE** If a non-dictionary column is of the **int** data type, it is internally stored as the **BigInt** type.
String	100,000 characters **NOTE** If the **CHAR** or **VARCHAR** data type is used in **CREATE TABLE**, the two data types are automatically converted to the String data type. If a column contains more than 32,000 characters, add the column to the **LONG_STRING_COLUMNS** attribute of the **tblproperties** table during table creation.
BigInt	64-bit value ranging from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
SmallInt	–32,768 to 32,767
Char	A to Z and a to z
Varchar	A to Z, a to z, and 0 to 9
Boolean	**true** or **false**
Decimal	The default value is (10,0) and maximum value is (38,38). **NOTE** When query with filters, append **BD** to the number to achieve accurate results. For example, **select * from carbon_table where num = 1234567890123456.22BD**.
Double	64-bit value ranging from 4.9E-324 to 1.7976931348623157E308
TimeStamp	The default format is **yyyy-MM-dd HH:mm:ss**.
Date	The **DATE** data type is used to store calendar dates. The default format is **yyyy-MM-DD**.

Data Type	Value Range
Array<data_type>	N/A **NOTE** Currently, only two layers of complex types can be nested.
Struct<col_name: data_type COMMENT col_comment, ...>	
Map<primitive_type, data_type>	

12.3.3.2.2 Creating a CarbonData Table

Scenario

A CarbonData table must be created to load and query data. You can run the **Create Table** command to create a table. This command is used to create a table using custom columns.

Creating a Table with Self-Defined Columns

Users can create a table by specifying its columns and data types.

Sample command:

CREATE TABLE *IF NOT EXISTS productdb.productSalesTable (*

productNumber Int,

productName String,

storeCity String,

storeProvince String,

productCategory String,

productBatch String,

saleQuantity Int,

revenue Int)

STORED AS *carbondata*

TBLPROPERTIES (

'table_blocksize'='128');

The following table describes parameters of preceding commands.

Table 12-13 Parameter description

Parameter	Description
productSalesTable	Table name. The table is used to load data for analysis. The table name consists of letters, digits, and underscores (_).
productdb	Database name. The database maintains logical connections with tables stored in it to identify and manage the tables. The database name consists of letters, digits, and underscores (_).
productName storeCity storeProvince procuctCategory productBatch saleQuantity revenue	Columns in the table. The columns are service entities for data analysis. The column name (field name) consists of letters, digits, and underscores (_).
table_blocksize	Indicates the block size of data files used by the CarbonData table, in MB. The value ranges from **1** to **2048**. The default value is **1024**. If **table_blocksize** is too small, a large number of small files will be generated when data is loaded. This may affect the performance of HDFS. If **table_blocksize** is too large, during data query, the amount of block data that matches the index is large, and some blocks contain a large number of blocklets, affecting read concurrency and lowering query performance. You are advised to set the block size based on the data volume. For example, set the block size to 256 MB for GB-level data, 512 MB for TB-level data, and 1024 MB for PB-level data.

NOTE

- Measurement of all Integer data is processed and displayed using the **BigInt** data type.
- CarbonData parses data strictly. Any data that cannot be parsed is saved as **null** in the table. For example, if the user loads the **double** value (3.14) to the BigInt column, the data is saved as **null**.
- The Short and Long data types used in the **Create Table** command are shown as Smallint and BigInt in the **DESCRIBE** command, respectively.
- You can run the **DESCRIBE** command to view the table data size and table index size.

Operation Result

Run the command to create a table.

12.3.3.2.3 Deleting a CarbonData Table

Scenario

You can run the **DROP TABLE** command to delete a table. After a CarbonData table is deleted, its metadata and loaded data are deleted together.

Procedure

Run the following command to delete a CarbonData table:

Run the following command:

DROP TABLE *[IF EXISTS] [db_name.]table_name;*

Once this command is executed, the table is deleted from the system. In the command, **db_name** is an optional parameter. If **db_name** is not specified, the table named **table_name** in the current database is deleted.

Example:

DROP TABLE *productdb.productSalesTable;*

Run the preceding command to delete the **productSalesTable** table from the **productdb** database.

Operation Result

Deletes the table specified in the command from the system. After the table is deleted, you can run the **SHOW TABLES** command to check whether the table is successfully deleted. For details, see **SHOW TABLES**.

12.3.3.2.4 Modify the CarbonData Table

SET and UNSET

When the **SET** command is executed, the new properties overwrite the existing ones.

- SORT SCOPE

  The following is an example of the **SET SORT SCOPE** command:

  ALTER TABLE *tablename* **SET TBLPROPERTIES('SORT_SCOPE'**=*'no_sort'*)

  After running the **UNSET SORT SCOPE** command, the default value **NO_SORT** is adopted.

  The following is an example of the **UNSET SORT SCOPE** command:

  ALTER TABLE *tablename* **UNSET TBLPROPERTIES('SORT_SCOPE'**)
- SORT COLUMNS

  The following is an example of the **SET SORT COLUMNS** command:

  ALTER TABLE *tablename* **SET TBLPROPERTIES('SORT_COLUMNS'**=*'column1'*)

After this command is executed, the new value of **SORT_COLUMNS** is used. Users can adjust the **SORT_COLUMNS** based on the query results, but the original data is not affected. The operation does not affect the query performance of the original data segments which are not sorted by new **SORT_COLUMNS**.

The **UNSET** command is not supported, but the **SORT_COLUMNS** can be set to empty string instead of using the **UNSET** command.

ALTER TABLE *tablename* **SET TBLPROPERTIES('SORT_COLUMNS'**= '')

NOTE

- The later version will enhance custom compaction to resort the old segments.
- The value of **SORT_COLUMNS** cannot be modified in the streaming table.
- If the **inverted index** column is removed from **SORT_COLUMNS**, **inverted index** will not be created in this column. However, the old configuration of **INVERTED_INDEX** will be kept.

12.3.3.3 CarbonData Table Data Management

12.3.3.3.1 Loading Data

Scenario

After a CarbonData table is created, you can run the **LOAD DATA** command to load data to the table for query. Once data loading is triggered, data is encoded in CarbonData format and files in multi-dimensional and column-based format are compressed and copied to the HDFS path of CarbonData files for quick analysis and queries. The HDFS path can be configured in the **carbon.properties** file. For details, see **Configuration Reference**.

12.3.3.3.2 Deleting Segments

Scenario

If you want to modify and reload the data because you have loaded wrong data into a table, or there are too many bad records, you can delete specific segments by segment ID or data loading time.

NOTE

The segment deletion operation only deletes segments that are not compacted. You can run the **CLEAN FILES** command to clear compacted segments.

Deleting a Segment by Segment ID

Each segment has a unique ID. This segment ID can be used to delete the segment.

Step 1 Obtain the segment ID.

Command:

SHOW SEGMENTS FOR Table *dbname.tablename LIMIT number_of_loads;*

Example:

SHOW SEGMENTS FOR TABLE *carbonTable;*

Run the preceding command to show all the segments of the table named **carbonTable**.

SHOW SEGMENTS FOR TABLE *carbonTable LIMIT 2;*

Run the preceding command to show segments specified by *number_of_loads*.

The command output is as follows:

```
+-----+----------+--------------------------+------------------+------------+------------+-------------+--------------+--
+
| ID  |  Status  |     Load Start Time      | Load Time Taken  | Partition  | Data Size  | Index Size  | File Format  |
+-----+----------+--------------------------+------------------+------------+------------+-------------+--------------+--
+
| 3   | Success  | 2020-09-28 22:53:26.336  | 3.726S           | {}         | 6.47KB     | 3.30KB      | columnar_v3  |
| 2   | Success  | 2020-09-28 22:53:01.702  | 6.688S           | {}         | 6.47KB     | 3.30KB      | columnar_v3  |
+-----+----------+--------------------------+------------------+------------+------------+-------------+--------------+--
+
```

NOTE

The output of the **SHOW SEGMENTS** command includes ID, Status, Load Start Time, Load Time Taken, Partition, Data Size, Index Size, and File Format. The latest loading information is displayed in the first line of the command output.

Step 2 Run the following command to delete the segment after you have found the Segment ID:

Command:

DELETE FROM TABLE tableName WHERE SEGMENT.ID IN (load_sequence_id1, load_sequence_id2, ...);

Example:

DELETE FROM TABLE carbonTable WHERE SEGMENT.ID IN (1,2,3);

For details, see **DELETE SEGMENT by ID**.

----End

Deleting a Segment by Data Loading Time

You can delete a segment based on the loading time.

Command:

DELETE FROM TABLE db_name.table_name WHERE SEGMENT.STARTTIME BEFORE date_value;

Example:

DELETE FROM TABLE carbonTable WHERE SEGMENT.STARTTIME BEFORE '2017-07-01 12:07:20';

The preceding command can be used to delete all segments before 2017-07-01 12:07:20.

For details, see **DELETE SEGMENT by DATE**.

Result

Data of corresponding segments is deleted and is unavailable for query. You can run the **SHOW SEGMENTS** command to display the segment status and check whether the segment has been deleted.

NOTE

- Segments are not physically deleted after the execution of the **DELETE SEGMENT** command. Therefore, if you run the **SHOW SEGMENTS** command to check the status of a deleted segment, it will be marked as **Marked for Delete**. If you run the **SELECT * FROM tablename** command, the deleted segment will be excluded.
- The deleted segment will be deleted physically only when the next data loading reaches the maximum query execution duration, which is configured by the **max.query.execution.time** parameter. The default value of the parameter is 60 minutes.
- If you want to forcibly delete a physical segment file, run the **CLEAN FILES** command.

  Example:

  CLEAN FILES FOR TABLE table1;

  This command will physically delete the segment file in the **Marked for delete** state.

  If this command is executed before the time specified by **max.query.execution.time** arrives, the query may fail. **max.query.execution.time** indicates the maximum time allowed for a query, which is set in the **carbon.properties** file.

12.3.3.3.3 Combining Segments

Scenario

Frequent data access results in a large number of fragmented CarbonData files in the storage directory. In each data loading, data is sorted and indexing is performed. This means that an index is generated for each load. With the increase of data loading times, the number of indexes also increases. As each index works only on one loading, the performance of index is reduced. CarbonData provides loading and compression functions. In a compression process, data in each segment is combined and sorted, and multiple segments are combined into one large segment.

Prerequisites

Multiple data loadings have been performed.

Operation Description

There are three types of compaction: Minor, Major, and Custom.

- Minor compaction:

  In minor compaction, you can specify the number of loads to be merged. If **carbon.enable.auto.load.merge** is set, minor compaction is triggered for every data load. If any segments are available to be merged, then compaction will run parallel with data load.

  There are two levels in minor compaction:

  – Level 1: Merging of the segments which are not yet compacted

  - Level 2: Merging of the compacted segments again to form a larger segment

- Major compaction:

  Multiple segments can be merged into one large segment. You can specify the compaction size so that all segments below the size will be merged. Major compaction is usually done during the off-peak time.

- Custom compaction:

  In Custom compaction, you can specify the IDs of multiple segments to merge them into a large segment. The IDs of all the specified segments must exist and be valid. Otherwise, the compaction fails. Custom compaction is usually done during the off-peak time.

For details, see **ALTER TABLE COMPACTION**.

Table 12-14 Compaction parameters

Parameter	Default Value	Application Type	Description
carbon.enable.auto.load.merge	false	Minor	Whether to enable compaction along with data loading. **true**: Compaction is automatically triggered when data is loaded. **false**: Compaction is not triggered when data is loaded.
carbon.compaction.level.threshold	4,3	Minor	This configuration is for minor compaction which decides how many segments to be merged. For example, if this parameter is set to **2,3**, minor compaction is triggered every two segments and segments form a single level 1 compacted segment. When the number of compacted level 1 segments reach 3, compaction is triggered again to merge them to form a single level 2 segment. The compaction policy depends on the actual data size and available resources. The value ranges from 0 to 100.

Parameter	Default Value	Application Type	Description
carbon.major.compaction.size	1024 MB	Major	The major compaction size can be configured using this parameter. Sum of the segments which is below this threshold will be merged. For example, if this parameter is set to 1024 MB, and there are five segments whose sizes are 300 MB, 400 MB, 500 MB, 200 MB, and 100 MB used for major compaction, only segments whose total size is less than this threshold are compacted. In this example, only the segments whose sizes are 300 MB, 400 MB, 200 MB, and 100 MB are compacted.
carbon.numberof.preserve.segments	0	Minor/ Major	If you want to preserve some number of segments from being compacted, then you can set this configuration. For example, if **carbon.numberof.preserve.segments** is set to **2**, the latest two segments will always be excluded from the compaction. By default, no segments are reserved.
carbon.allowed.compaction.days	0	Minor/ Major	This configuration is used to control on the number of recent segments that needs to be compacted. For example, if this parameter is set to **2**, the segments which are loaded in the time frame of past 2 days only will get merged. Segments which are loaded earlier than 2 days will not be merged. This configuration is disabled by default.
carbon.number.of.cores.while.compacting	2	Minor/ Major	Number of cores to be used while compacting data. The greater the number of cores, the better the compaction performance. If the CPU resources are sufficient, you can increase the value of this parameter.

Parameter	Default Value	Application Type	Description
carbon.merge.index.in.segment	true	SEGMENT_INDEX	If this parameter is set to **true**, all the Carbon index (.carbonindex) files in a segment will be merged into a single Index (.carbonindexmerge) file. This enhances the first query performance.

Reference

You are advised not to perform minor compaction on historical data. For details, see **How to Avoid Minor Compaction for Historical Data?**.

12.3.3.4 CarbonData Data Migration

Scenario

If you want to rapidly migrate CarbonData data from a cluster to another one, you can use the CarbonData backup and restoration commands. This method does not require data import in the target cluster, reducing required migration time.

Prerequisites

The Spark2x client has been installed in a directory, for example, **/opt/client**, in two clusters. The source cluster is cluster A, and the target cluster is cluster B.

Procedure

Step 1 Log in to the node where the client is installed in cluster A as a client installation user.

Step 2 Run the following commands to configure environment variables:

source /opt/client/bigdata_env

source /opt/client/Spark2x/component_env

Step 3 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, skip user authentication.

kinit *carbondatauser*

carbondatauser indicates the user of the original data. That is, the user has the read and write permissions for the tables.

NOTE

You must add the user to the **hadoop** (primary group) and **hive** groups, and associate it with the **System_administrator** role.

Step 4 Run the following command to connect to the database and check the location for storing table data on HDFS:

spark-beeline

desc formatted *Name of the table containing the original data*;

Location in the displayed information indicates the directory where the data file resides.

Step 5 Log in to the node where the client is installed in cluster B as a client installation user and configure the environment variables:

source /opt/client/bigdata_env

source /opt/client/Spark2x/component_env

Step 6 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, skip user authentication.

kinit *carbondatauser2*

carbondatauser2 indicates the user that uploads data.

NOTE

You must add the user to the **hadoop** (primary group) and **hive** groups, and associate it with the **System_administrator** role.

Step 7 Run the **spark-beeline** command to connect to the database.

Step 8 Does the database that maps to the original data exist?

- If yes, go to **Step 9**.
- If no, run the **create database** *Database name* command to create a database with the same name as that maps to the original data and go to **Step 9**.

Step 9 Copy the original data from the HDFS directory in cluster A to that in cluster B.

When uploading data in cluster B, ensure that the upload directory has the directories with the same names as the database and table in the original directory and the upload user has the permission to write data to the upload directory. After the data is uploaded, the user has the permission to read and write the data.

For example, if the original data is stored in **/user/carboncadauser/warehouse/db1/tb1**, the data can be stored in **/user/carbondatauser2/warehouse/db1/tb1** in the new cluster.

1. Run the following command to download the original data to the **/opt/backup** directory of cluster A:

   hdfs dfs -get /user/carboncadauser/warehouse/db1/tb1 /opt/backup

2. Run the following command to copy the original data of cluster A to the **/opt/backup** directory on the client node of cluster B.

   scp /opt/backup root@*IP address of the client node of cluster B*:**/opt/backup**

3. Run the following command to upload the data copied to cluster B to HDFS:

   hdfs dfs -put /opt/backup /user/carbondatauser2/warehouse/db1/tb1

Step 10 In the client environment of cluster B, run the following command to generate the metadata associated with the table corresponding to the original data in Hive:

REFRESH TABLE *$dbName.$tbName;*

$dbName indicates the database name, and *$tbName* indicates the table name.

Step 11 If the original table contains an index table, perform **Step 9** and **Step 10** to migrate the index table directory from cluster A to cluster B.

Step 12 Run the following command to register an index table for the CarbonData table (skip this step if no index table is created for the original table):

REGISTER INDEX TABLE *$tableName* ON *$maintable*;

$tableName indicates the index table name, and *$maintable* indicates the table name.

----End

12.3.3.5 Migrating Data on CarbonData from Spark 1.5 to Spark2x

Migration Solution Overview

This migration guides you to migrate the CarbonData table data of Spark 1.5 to that of Spark2x.

NOTE

Before performing this operation, you need to stop the data import service of the CarbonData table in Spark 1.5 and migrate data to the CarbonData table of Spark2x at a time. After the migration is complete, use Spark2x to perform service operations.

Migration roadmap:

1. Use Spark 1.5 to migrate historical data to the intermediate table.
2. Use Spark2x to migrate data from the intermediate table to the target table and change the target table name to the original table name.
3. After the migration is complete, use Spark2x to operate data in the CarbonData table.

Migration Solution and Commands

Migrating Historical Data

Step 1 Stop the CarbonData data import service, use spark-beeline of Spark 1.5 to view the ID and time of the latest segment in the CarbonData table, and record the segment ID.

show segments for table dbname.tablename;

Step 2 Run spark-beeline of Spark 1.5 as the user who has created the original CarbonData table to create an intermediate table in ORC or Parquet format. Then import the data in the original CarbonData table to the intermediate table. After the import is complete, the services of the CarbonData table can be restored.

Create an ORC table.

CREATE TABLE dbname.mid_tablename_orc STORED AS ORC as select * from dbname.tablename;

Create a Parquet table.

CREATE TABLE dbname.mid_tablename_parq STORED AS PARQUET as select * from dbname.tablename;

In the preceding command, **dbname** indicates the database name and **tablename** indicates the name of the original CarbonData table.

Step 3 Run spark-beeline of Spark2x as the user who has created the original CarbonData table. Run the table creation statement of the old table to create a CarbonData table.

NOTE

In the statement for creating a new table, the field sequence and type must be the same as those of the old table. In this way, the index column structure of the old table can be retained, which helps avoid errors caused by the use of **select *** statement during data insertion.

Run the spark-beeline command of Spark 1.5 to view the table creation statement of the old table: **SHOW CREATE TABLE dbname.tablename;**

Create a CarbonData table named **dbname.new_tablename**.

Step 4 Run spark-beeline of Spark2x as the user who has created the original CarbonData table to load the intermediate table data in ORC (or PARQUET) format created in **Step 2** to the new table created in **Step 3**. This step may take a long time (about 2 hours for 200 GB data). The following uses the ORC intermediate table as an example to describe the command for loading data:

insert into dbname.new_tablename select *

from dbname. mid_tablename_orc;

Step 5 Run spark-beeline of Spark2x as the user who has created the original CarbonData table to query and verify the data in the new table. If the data is correct, change the name of the original CarbonData table and then change the name of the new CarbonData table to the name of the original one.

ALTER TABLE dbname.tablename RENAME TO dbname.old_tablename;

ALTER TABLE dbname.new_tablename RENAME TO dbname.tablename;

Step 6 Complete the migration. In this case, you can use Spark2x to query the new table and rebuild the secondary index.

----End

12.3.4 CarbonData Performance Tuning

12.3.4.1 Tuning Guidelines

Query Performance Tuning

There are various parameters that can be tuned to improve the query performance in CarbonData. Most of the parameters focus on increasing the parallelism in processing and optimizing system resource usage.

- Spark executor count: Executors are basic entities of parallelism in Spark. Raising the number of executors can increase the amount of parallelism in the cluster. For details about how to configure the number of executors, see the Spark documentation.
- Executor core: The number of concurrent tasks that an executor can run are controlled in each executor. Increasing the number of executor cores will add more concurrent processing tasks to improve performance.
- HDFS block size: CarbonData assigns query tasks by allocating different blocks to different executors for processing. HDFS block is the partition unit. CarbonData maintains a global block level index in Spark driver, which helps to reduce the quantity of blocks that need to be scanned for a query. Higher block size means higher I/O efficiency and lower global index efficiency. Reversely, lower block size means lower I/O efficiency, higher global index efficiency, and greater memory consumption.
- Number of scanner threads: Scanner threads control the number of parallel data blocks that are processed by each task. By increasing the number of scanner threads, you can increase the number of data blocks that are processed in parallel to improve performance. The **carbon.number.of.cores** parameter in the **carbon.properties** file is used to configure the number of scanner threads. For example, **carbon.number.of.cores = 4**.
- B-Tree caching: The cache memory can be optimized using the B-Tree least recently used (LRU) caching. In the driver, the B-Tree LRU caching configuration helps free up the cache by releasing table segments which are not accessed or not used. Similarly, in the executor, the B-Tree LRU caching configuration will help release table blocks that are not accessed or used. For details, see the description of **carbon.max.driver.lru.cache.size** and **carbon.max.executor.lru.cache.size** in **Table 12-6**.

CarbonData Query Process

When CarbonData receives a table query task, for example query for table A, the index data of table A will be loaded to the memory for the query process. When CarbonData receives a query task for table A again, the system does not need to load the index data of table A.

When a query is performed in CarbonData, the query task is divided into several scan tasks, namely, task splitting based on HDFS blocks. Scan tasks are executed by executors on the cluster. Tasks can run in parallel, partially parallel, or in sequence, depending on the number of executors and configured number of executor cores.

Some parts of a query task can be processed at the individual task level, such as **select** and **filter.** Some parts of a query task can be processed at the individual task level, such as **group-by**, **count**, and **distinct count**.

Some operations cannot be performed at the task level, such as **Having Clause** (filter after grouping) and **sort**. Operations which cannot be performed at the task level or can be only performed partially at the task level require data (partial results) transmission across executors on the cluster. The transmission operation is called shuffle.

The more the tasks are, the more data needs to be shuffled. This affects query performance.

The number of tasks is depending on the number of HDFS blocks and the number of blocks is depending on the size of each block. You are advised to configure proper HDFS block size to achieve a balance among increased parallelism, the amount of data to be shuffled, and the size of aggregate tables.

Relationship Between Splits and Executors

If the number of splits is less than or equal to the executor count multiplied by the executor core count, the tasks are run in parallel. Otherwise, some tasks can start only after other tasks are complete. Therefore, ensure that the executor count multiplied by executor cores is greater than or equal to the number of splits. In addition, make sure that there are sufficient splits so that a query task can be divided into sufficient subtasks to ensure concurrency.

Configuring Scanner Threads

The scanner threads property decides the number of data blocks to be processed. If there are too many data blocks, a large number of small data blocks will be generated, affecting performance. If there are few data blocks, the parallelism is poor and the performance is affected. Therefore, when determining the number of scanner threads, you are advised to consider the average data size within a partition and select a value that makes the data block not small. Based on experience, you are advised to divide a single block size (unit: MB) by 250 and use the result as the number of scanner threads.

The number of actual available vCPUs is an important factor to consider when you want to increase the parallelism. The number of vCPUs that conduct parallel computation must not exceed 75% to 80% of actual vCPUs.

The number of vCPUs is approximately equal to:

Number of parallel tasks x Number of scanner threads. Number of parallel tasks is the smaller value of number of splits or executor count x executor cores.

Data Loading Performance Tuning

Tuning of data loading performance is different from that of query performance. Similar to query performance, data loading performance depends on the amount of parallelism that can be achieved. In case of data loading, the number of worker threads decides the unit of parallelism. Therefore, more executors mean more executor cores and better data loading performance.

To achieve better performance, you can configure the following parameters in HDFS.

Table 12-15 HDFS configuration

Parameter	Recommended Value
dfs.datanode.drop.cache.behind.reads	false
dfs.datanode.drop.cache.behind.writes	false
dfs.datanode.sync.behind.writes	true

Compression Tuning

CarbonData uses a few lightweight compression and heavyweight compression algorithms to compress data. Although these algorithms can process any type of data, the compression performance is better if the data is ordered with similar values being together.

During data loading, data is sorted based on the order of columns in the table to achieve good compression performance.

Since CarbonData sorts data in the order of columns defined in the table, the order of columns plays an important role in the effectiveness of compression. If the low cardinality dimension is on the left, the range of data partitions after sorting is small and the compression efficiency is high. If a high cardinality dimension is on the left, a range of data partitions obtained after sorting is relatively large, and compression efficiency is relatively low.

Memory Tuning

CarbonData provides a mechanism for memory tuning where data loading depends on the columns needed in the query. Whenever a query command is received, columns required by the query are fetched and data is loaded for those columns in memory. During this operation, if the memory threshold is reached, the least used loaded files are deleted to release memory space for columns required by the query.

12.3.4.2 Suggestions for Creating CarbonData Tables

Scenario

This section provides suggestions based on more than 50 test cases to help you create CarbonData tables with higher query performance.

Table 12-16 Columns in the CarbonData table

Column name	Data type	Cardinality	Attribution
msisdn	String	30 million	dimension
BEGIN_TIME	bigint	10,000	dimension
host	String	1 million	dimension

Column name	Data type	Cardinality	Attribution
dime_1	String	1,000	dimension
dime_2	String	500	dimension
dime_3	String	800	dimension
counter_1	numeric(20,0)	NA	measure
...	...	NA	measure
counter_100	numeric(20,0)	NA	measure

Procedure

- If the to-be-created table contains a column that is frequently used for filtering, for example, this column is used in more than 80% of filtering scenarios,

  implement optimization as follows:

  Place this column in the first column of **sort_columns**.

  For example, if **msisdn** is the most frequently used filter criterion in a query, it is placed in the first column. Run the following command to create a table. The query performance is good if **msisdn** is used as the filter condition.

  ```
  create table carbondata_table(
     msisdn String,
     ...
     )STORED AS carbondata TBLPROPERTIES ('SORT_COLUMS'='msisdn');
  ```

- If the to-be-created table has multiple columns which are frequently used to filter the results,

  implement optimization as follows:

  Create an index for the columns.

  For example, if **msisdn**, **host**, and **dime_1** are frequently used columns, the **sort_columns** column sequence is "dime_1-> host-> msisdn..." based on cardinality. Run the following command to create a table. The following command can improve the filtering performance of **dime_1**, **host**, and **msisdn**.

  ```
  create table carbondata_table(
     dime_1 String,
     host String,
     msisdn String,
     dime_2 String,
     dime_3 String,
     ...
     )STORED AS carbondata
  TBLPROPERTIES ('SORT_COLUMS'='dime_1,host,msisdn');
  ```

- If the frequency of each column used for filtering is similar,

  implement optimization as follows:

  sort_columns is sorted in ascending order of cardinality.

  Run the following command to create a table:

  ```
  create table carbondata_table(
     Dime_1 String,
  ```

```
    BEGIN_TIME bigint,
    HOST String,
    MSISDN String,
    ...
    )STORED AS carbondata
TBLPROPERTIES ('SORT_COLUMS'='dime_2,dime_3,dime_1, BEGIN_TIME,host,msisdn');
```

- Create tables in ascending order of cardinalities. Then create secondary indexes for columns with more cardinalities. The statement for creating an index is as follows:

```
create index carbondata_table_index_msidn on tablecarbondata_table (
MSISDN String) as 'carbondata' PROPERTIES ('table_blocksize'='128');
create index carbondata_table_index_host on tablecarbondata_table (
host String) as 'carbondata' PROPERTIES ('table_blocksize'='128');
```

- For columns of measure type, not requiring high accuracy, the numeric (20,0) data type is not required. You are advised to use the double data type to replace the numeric (20,0) data type to enhance query performance.

  The result of performance analysis of test-case shows reduction in query execution time from 15 to 3 seconds, thereby improving performance by nearly 5 times. The command for creating a table is as follows:

```
create table carbondata_table(
    Dime_1 String,
    BEGIN_TIME bigint,
    HOST String,
    MSISDN String,
    counter_1 double,
    counter_2 double,
    ...
    counter_100 double,
    )STORED AS carbondata
;
```

- If values (**start_time** for example) of a column are incremental:

  For example, if data is loaded to CarbonData every day, **start_time** is incremental for each load. In this case, it is recommended that the **start_time** column be put at the end of **sort_columns**, because incremental values are efficient in using min/max index. The command for creating a table is as follows:

```
create table carbondata_table(
    Dime_1 String,
    HOST String,
    MSISDN String,
    counter_1 double,
    counter_2 double,
    BEGIN_TIME bigint,
    ...
    counter_100 double,
    )STORED AS carbondata
    TBLPROPERTIES ( 'SORT_COLUMS'='dime_2,dime_3,dime_1..BEGIN_TIME');
```

12.3.4.3 Configurations for Performance Tuning

Scenario

This section describes the configurations that can improve CarbonData performance.

Procedure

Table 12-17 and **Table 12-18** describe the configurations about query of CarbonData.

Table 12-17 Number of tasks started for the shuffle process

Parameter	spark.sql.shuffle.partitions
Configuration File	spark-defaults.conf
Function	Data query
Scenario Description	Number of tasks started for the shuffle process in Spark
Tuning	You are advised to set this parameter to one to two times as much as the executor cores. In an aggregation scenario, reducing the number from 200 to 32 can reduce the query time by two folds.

Table 12-18 Number of executors and vCPUs, and memory size used for CarbonData data query

Parameter	spark.executor.cores spark.executor.instances spark.executor.memory
Configuration File	spark-defaults.conf
Function	Data query
Scenario Description	Number of executors and vCPUs, and memory size used for CarbonData data query
Tuning	In the bank scenario, configuring 4 vCPUs and 15 GB memory for each executor will achieve good performance. The two values do not mean the more the better. Configure the two values properly in case of limited resources. If each node has 32 vCPUs and 64 GB memory in the bank scenario, the memory is not sufficient. If each executor has 4 vCPUs and 12 GB memory, Garbage Collection may occur during query, time spent on query from increases from 3s to more than 15s. In this case, you need to increase the memory or reduce the number of vCPUs.

Table 12-19, **Table 12-20**, and **Table 12-21** describe the configurations for CarbonData data loading.

Table 12-19 Number of vCPUs used for data loading

Parameter	carbon.number.of.cores.while.loading

Configuration File	carbon.properties
Function	Data loading
Scenario Description	Number of vCPUs used for data processing during data loading in CarbonData
Tuning	If there are sufficient CPUs, you can increase the number of vCPUs to improve performance. For example, if the value of this parameter is changed from 2 to 4, the CSV reading performance can be doubled.

Table 12-20 Whether to use Yarn local directories for multi-disk data loading

Parameter	carbon.use.local.dir
Configuration File	carbon.properties
Function	Data loading
Scenario Description	Whether to use Yarn local directories for multi-disk data loading
Tuning	If this parameter is set to **true**, CarbonData uses local Yarn directories for multi-table load disk load balance, improving data loading performance.

Table 12-21 Whether to use multiple directories during loading

Parameter	carbon.use.multiple.temp.dir
Configuration File	carbon.properties
Function	Data loading
Scenario Description	Whether to use multiple temporary directories to store temporary sort files
Tuning	If this parameter is set to **true**, multiple temporary directories are used to store temporary sort files during data loading. This configuration improves data loading performance and prevents single points of failure (SPOFs) on disks.

Table 12-22 describes the configurations for CarbonData data loading and query.

Table 12-22 Number of vCPUs used for data loading and query

Parameter	carbon.compaction.level.threshold
Configuration File	carbon.properties
Function	Data loading and query
Scenario Description	For minor compaction, specifies the number of segments to be merged in stage 1 and number of compacted segments to be merged in stage 2.
Tuning	Each CarbonData load will create one segment, if every load is small in size, it will generate many small files over a period of time impacting the query performance. Configuring this parameter will merge the small segments to one big segment which will sort the data and improve the performance. The compaction policy depends on the actual data size and available resources. For example, a bank loads data once a day and at night when no query is performed. If resources are sufficient, the compaction policy can be 6 or 5.

Table 12-23 Whether to enable data pre-loading when the index cache server is used

Parameter	carbon.indexserver.enable.prepriming
Configuration File	carbon.properties
Function	Data loading
Scenario Description	Enabling data pre-loading during the use of the index cache server can improve the performance of the first query.
Tuning	You can set this parameter to **true** to enable the pre-loading function. The default value is **false**.

12.3.5 CarbonData Access Control

The following table provides details about Hive ACL permissions required for performing operations on CarbonData tables.

Prerequisites

Parameters listed in **Table 12-9** or **Table 12-10** have been configured.

Hive ACL permissions

Table 12-24 Hive ACL permissions required for CarbonData table-level operations

Scenario	Required Permission
DESCRIBE TABLE	SELECT (of table)
SELECT	SELECT (of table)
EXPLAIN	SELECT (of table)
CREATE TABLE	CREATE (of database)
CREATE TABLE As SELECT	CREATE (on database), INSERT (on table), RW on data file, and SELECT (on table)
LOAD	INSERT (of table) RW on data file
DROP TABLE	OWNER (of table)
DELETE SEGMENTS	DELETE (of table)
SHOW SEGMENTS	SELECT (of table)
CLEAN FILES	DELETE (of table)
INSERT OVERWRITE / INSERT INTO	INSERT (of table) RW on data file and SELECT (of table)
CREATE INDEX	OWNER (of table)
DROP INDEX	OWNER (of table)
SHOW INDEXES	SELECT (of table)
ALTER TABLE ADD COLUMN	OWNER (of table)
ALTER TABLE DROP COLUMN	OWNER (of table)
ALTER TABLE CHANGE DATATYPE	OWNER (of table)
ALTER TABLE RENAME	OWNER (of table)
ALTER TABLE COMPACTION	INSERT (on table)
FINISH STREAMING	OWNER (of table)
ALTER TABLE SET STREAMING PROPERTIES	OWNER (of table)
ALTER TABLE SET TABLE PROPERTIES	OWNER (of table)
UPDATE CARBON TABLE	UPDATE (of table)
DELETE RECORDS	DELETE (of table)
REFRESH TABLE	OWNER (of main table)

Scenario	Required Permission
REGISTER INDEX TABLE	OWNER (of table)
SHOW PARTITIONS	SELECT (on table)
ALTER TABLE ADD PARTITION	OWNER (of table)
ALTER TABLE DROP PARTITION	OWNER (of table)

NOTE

- If tables in the database are created by multiple users, the **Drop database** command fails to be executed even if the user who runs the command is the owner of the database.
- In a secondary index, when the parent table is triggered, **insert** and **compaction** are triggered on the index table. If you select a query that has a filter condition that matches index table columns, you should provide selection permissions for the parent table and index table.
- The LockFiles folder and lock files created in the LockFiles folder will have full permissions, as the LockFiles folder does not contain any sensitive data.
- If you are using ACL, ensure you do not configure any path for DDL or DML which is being used by other process. You are advised to create new paths.

  Configure the path for the following configuration items:

  1) carbon.badRecords.location

  2) Db_Path and other items during database creation
- For Carbon ACL in a non-security cluster, **hive.server2.enable.doAs** in the **hive-site.xml** file must be set to **false**. Then the query will run as the user who runs the hiveserver2 process.

12.3.6 CarbonData Syntax Reference

12.3.6.1 DDL

12.3.6.1.1 CREATE TABLE

Function

This command is used to create a CarbonData table by specifying the list of fields along with the table properties.

Syntax

CREATE TABLE *[IF NOT EXISTS] [db_name.]table_name*

[(col_name data_type, ...)]

STORED AS *carbondata*

[TBLPROPERTIES (property_name=property_value, ...)];

Additional attributes of all tables are defined in **TBLPROPERTIES**.

Parameter Description

Table 12-25 CREATE TABLE parameters

Parameter	Description
db_name	Database name that contains letters, digits, and underscores (_).
col_name data_type	List with data types separated by commas (,). The column name contains letters, digits, and underscores (_). **NOTE** When creating a CarbonData table, do not use tupleId, PositionId, and PositionReference as column names because columns with these names are internally used by secondary index commands.
table_name	Table name of a database that contains letters, digits, and underscores (_).
STORED AS	The **carbondata** parameter defines and creates a CarbonData table.
TBLPROPERTIES	List of CarbonData table properties.

Precautions

Table attributes are used as follows:

- Block size

  The block size of a data file can be defined for a single table using **TBLPROPERTIES**. The larger one between the actual size of the data file and the defined block size is selected as the actual block size of the data file in HDFS. The unit is MB. The default value is 1024 MB. The value ranges from 1 MB to 2048 MB. If the value is beyond the range, the system reports an error.

  Once the block size reaches the configured value, the write program starts a new block of CarbonData data. Data is written in multiples of the page size (32,000 records). Therefore, the boundary is not strict at the byte level. If the new page crosses the boundary of the configured block, the page is written to the new block instead of the current block.

  TBLPROPERTIES('table_blocksize'='128')

  NOTE

  - If a small block size is configured in the CarbonData table while the size of the data file generated by the loaded data is large, the block size displayed in HDFS is different from the configured value. This is because when data is written to a local block file for the first time, even though the size of the to-be-written data is larger than the configured value of the block size, data will still be written into the block. Therefore, the actual value of block size in HDFS is the larger value between the size of the data to be written and the configured block size.
  - If **block.num** is less than the parallelism, the blocks are split into new blocks so that new blocks.num is greater than parallelism and all cores can be used. This optimization is called block distribution.

- **SORT_SCOPE** specifies the sort scope during table creation. There are four types of sort scopes:
  - **GLOBAL_SORT**: It improves query performance, especially for point queries. *TBLPROPERTIES('SORT_SCOPE'='GLOBAL_SORT'*)
  - **LOCAL_SORT**: Data is sorted locally (task-level sorting).
  - **NO_SORT**: The default sorting mode is used. Data is loaded in unsorted manner, which greatly improves loading performance.
- SORT_COLUMNS

  This table property specifies the order of sort columns.

  TBLPROPERTIES('SORT_COLUMNS'='column1, column3')

  NOTE

  - If this attribute is not specified, no columns are sorted by default.
  - If this property is specified but with empty argument, then the table will be loaded without sort. For example, *('SORT_COLUMNS'='')*.
  - **SORT_COLUMNS** supports the string, date, timestamp, short, int, long, byte, and boolean data types.
- RANGE_COLUMN

  This property is used to specify a column to partition the input data by range. Only one column can be configured. During data import, you can use **global_sort_partitions** or **scale_factor** to avoid generating small files.

  TBLPROPERTIES('RANGE_COLUMN'='column1')
- LONG_STRING_COLUMNS

  The length of a common string cannot exceed 32,000 characters. To store a string of more than 32,000 characters, set **LONG_STRING_COLUMNS** to the target column.

  TBLPROPERTIES('LONG_STRING_COLUMNS'='column1, column3')

  NOTE

  LONG_STRING_COLUMNS can be set only for columns of the STRING, CHAR, or VARCHAR type.

Scenarios

Creating a Table by Specifying Columns

The **CREATE TABLE** command is the same as that of Hive DDL. The additional configurations of CarbonData are provided as table properties.

CREATE TABLE *[IF NOT EXISTS] [db_name.]table_name*

[(col_name data_type , ...)]

STORED AS *carbondata*

[TBLPROPERTIES (property_name=property_value, ...)];

Examples

CREATE TABLE *IF NOT EXISTS productdb.productSalesTable (*

productNumber Int,

productName String,

storeCity String,

storeProvince String,

productCategory String,

productBatch String,

saleQuantity Int,

revenue Int)

STORED AS carbondata

TBLPROPERTIES (

'table_blocksize'='128',

'SORT_COLUMNS'='productBatch, productName')

System Response

A table will be created and the success message will be logged in system logs.

12.3.6.1.2 CREATE TABLE As SELECT

Function

This command is used to create a CarbonData table by specifying the list of fields along with the table properties.

Syntax

CREATE TABLE *[IF NOT EXISTS] [db_name.]table_name* ***STORED AS carbondata*** *[TBLPROPERTIES (key1=val1, key2=val2, ...)] AS select_statement;*

Parameter Description

Table 12-26 CREATE TABLE parameters

Parameter	Description
db_name	Database name that contains letters, digits, and underscores (_).
table_name	Table name of a database that contains letters, digits, and underscores (_).
STORED AS	Used to store data in CarbonData format.
TBLPROPERTIES	List of CarbonData table properties. For details, see **Precautions**.

Precautions

N/A

Examples

CREATE TABLE ctas_select_parquet ***STORED AS*** carbondata as select * from parquet_ctas_test;

System Response

This example will create a Carbon table from any Parquet table and load all the records from the Parquet table.

12.3.6.1.3 DROP TABLE

Function

This command is used to delete an existing table.

Syntax

DROP TABLE *[IF EXISTS] [db_name.]table_name;*

Parameter Description

Table 12-27 DROP TABLE parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is selected.
table_name	Name of the table to be deleted

Precautions

In this command, **IF EXISTS** and **db_name** are optional.

Example

DROP TABLE IF EXISTS productDatabase.productSalesTable;

System Response

The table will be deleted.

12.3.6.1.4 SHOW TABLES

Function

SHOW TABLES command is used to list all tables in the current or a specific database.

Syntax

SHOW TABLES *[IN db_name];*

Parameter Description

Table 12-28 SHOW TABLE parameters

Parameter	Description
IN db_name	Name of the database. This parameter is required only when tables of this specific database are to be listed.

Usage Guidelines

IN db_Name is optional.

Examples

SHOW TABLES IN ProductDatabase;

System Response

All tables are listed.

12.3.6.1.5 ALTER TABLE COMPACTION

Function

The **ALTER TABLE COMPACTION** command is used to merge a specified number of segments into a single segment. This improves the query performance of a table.

Syntax

ALTER TABLE*[db_name.]table_name COMPACT 'MINOR/MAJOR/ SEGMENT_INDEX';*

ALTER TABLE*[db_name.]table_name COMPACT 'CUSTOM' WHERE SEGMENT.ID IN (id1, id2, ...);*

Parameter Description

Table 12-29 ALTER TABLE COMPACTION parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is selected.
table_name	Table name.
MINOR	Minor compaction. For details, see **Combining Segments**.
MAJOR	Major compaction. For details, see **Combining Segments**.
SEGMENT_INDEX	This configuration enables you to merge all the CarbonData index files (**.carbonindex**) inside a segment to a single CarbonData index merge file (**.carbonindexmerge**). This enhances the first query performance. For more information, see **Table 12-14**.
CUSTOM	Custom compaction. For details, see **Combining Segments**.

Precautions

N/A

Examples

ALTER TABLE ProductDatabase COMPACT 'MINOR';

ALTER TABLE ProductDatabase COMPACT 'MAJOR';

ALTER TABLE ProductDatabase COMPACT 'SEGMENT_INDEX';

ALTER TABLE ProductDatabase COMPACT 'CUSTOM' WHERE SEGMENT.ID IN (0, 1);

System Response

ALTER TABLE COMPACTION does not show the response of the compaction because it is run in the background.

If you want to view the response of minor and major compactions, you can check the logs or run the ***SHOW SEGMENTS*** command.

Example:

```
+------+------------+--------------------------+------------------+------------+------------+-------------+--------------
+--+
|  ID  |   Status   |     Load Start Time      | Load Time Taken  | Partition  | Data Size  | Index Size  | File
Format  |
+------+------------+--------------------------+------------------+------------+------------+-------------+--------------
+--+
| 3    | Success    | 2020-09-28 22:53:26.336  | 3.726S           | {}         | 6.47KB     | 3.30KB      | columnar_v3  |
| 2    | Success    | 2020-09-28 22:53:01.702  | 6.688S           | {}         | 6.47KB     | 3.30KB      | columnar_v3  |
| 1    | Compacted  | 2020-09-28 22:51:15.242  | 5.82S            | {}         | 6.50KB     | 3.43KB      |
```

```
columnar_v3  |
| 0.1  | Success    | 2020-10-30 20:49:24.561  | 16.66S          | {}        | 12.87KB    | 6.91KB      | columnar_v3
|
| 0    | Compacted  | 2020-09-28 22:51:02.6    | 6.819S          | {}        | 6.50KB     | 3.43KB      | columnar_v3
|
+------+------------+--------------------------+-----------------+-----------+------------+-------------+-------------
+--+
```

In the preceding information:

- **Compacted** indicates that data has been compacted.
- **0.1** indicates the compacting result of segment 0 and segment 1.

The compact operation does not incur any change to other operations.

Compacted segments, such as segment 0 and segment 1, become useless. To save space, before you perform other operations, run the ***CLEAN FILES*** command to delete compacted segments. For more information about the **CLEAN FILES** command, see **CLEAN FILES**.

12.3.6.1.6 TABLE RENAME

Function

This command is used to rename an existing table.

Syntax

ALTER TABLE *[db_name.]table_name* **RENAME TO** *new_table_name*;

Parameter Description

Table 12-30 RENAME parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is selected.
table_name	Current name of the existing table
new_table_name	New name of the existing table

Precautions

- Parallel queries (using table names to obtain paths for reading CarbonData storage files) may fail during this operation.
- The secondary index table cannot be renamed.

Example

ALTER TABLE *carbon* **RENAME TO** *carbondata*;

ALTER TABLE *test_db.carbon* **RENAME TO** *test_db.carbondata*;

System Response

The new table name will be displayed in the CarbonData folder. You can run **SHOW TABLES** to view the new table name.

12.3.6.1.7 ADD COLUMNS

Function

This command is used to add a column to an existing table.

Syntax

ALTER TABLE *[db_name.]table_name* **ADD COLUMNS** *(col_name data_type,...)* **TBLPROPERTIES***(''COLUMNPROPERTIES.columnName.shared_column'='sharedFolder.sharedColumnName,...', 'DEFAULT.VALUE.COLUMN_NAME'='default_value')*;

Parameter Description

Table 12-31 ADD COLUMNS parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is selected.
table_name	Table name.
col_name data_type	Name of a comma-separated column with a data type. It consists of letters, digits, and underscores (_). **NOTE** When creating a CarbonData table, do not name columns as tupleId, PositionId, and PositionReference because they will be used in UPDATE, DELETE, and secondary index commands.

Precautions

- Only **shared_column** and **default_value** are read. If any other property name is specified, no error will be thrown and the property will be ignored.
- If no default value is specified, the default value of the new column is considered null.
- If filter is applied to the column, new columns will not be added during sort. New columns may affect query performance.

Examples

- **ALTER TABLE** *carbon* **ADD COLUMNS** *(a1 INT, b1 STRING)*;
- **ALTER TABLE** *carbon* **ADD COLUMNS** *(a1 INT, b1 STRING)* **TBLPROPERTIES***('COLUMNPROPERTIES.b1.shared_column'='sharedFolder.b1')*;
- ALTER TABLE *carbon* **ADD COLUMNS** *(a1 INT, b1 STRING)* **TBLPROPERTIES***('DEFAULT.VALUE.a1'='10')*;

System Response

The newly added column can be displayed by running the **DESCRIBE** command.

12.3.6.1.8 DROP COLUMNS

Function

This command is used to delete one or more columns from a table.

Syntax

ALTER TABLE *[db_name.]table_name* **DROP COLUMNS** *(col_name, ...)*;

Parameter Description

Table 12-32 DROP COLUMNS parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is selected.
table_name	Table name.
col_name	Name of a column in a table. Multiple columns are supported. It consists of letters, digits, and underscores (_).

Precautions

After a column is deleted, at least one key column must exist in the schema. Otherwise, an error message is displayed, and the column fails to be deleted.

Examples

Assume that the table contains four columns named a1, b1, c1, and d1.

- Delete a column:
  ALTER TABLE *carbon* **DROP COLUMNS** *(b1)*;
  ALTER TABLE *test_db.carbon* **DROP COLUMNS** *(b1)*;
- Delete multiple columns:
  ALTER TABLE *carbon* **DROP COLUMNS** *(b1,c1)*;
  ALTER TABLE *test_db.carbon* **DROP COLUMNS** *(b1,c1)*;

System Response

If you run the **DESCRIBE** command, the deleted columns will not be displayed.

12.3.6.1.9 CHANGE DATA TYPE

Function

This command is used to change the data type from INT to BIGINT or decimal precision from lower to higher.

Syntax

ALTER TABLE *[db_name.]table_name* **CHANGE** *col_name col_name changed_column_type*;

Parameter Description

Table 12-33 CHANGE DATA TYPE parameters

Parameter	Description
db_name	Name of the database. If this parameter is left unspecified, the current database is selected.
table_name	Name of the table.
col_name	Name of columns in a table. Column names contain letters, digits, and underscores (_).
changed_column_type	The change in the data type.

Usage Guidelines

- Change of decimal data type from lower precision to higher precision will only be supported for cases where there is no data loss.

  Example:

  - **Invalid scenario** - Change of decimal precision from (10,2) to (10,5) is not valid as in this case only scale is increased but total number of digits remain the same.
  - **Valid scenario** - Change of decimal precision from (10,2) to (12,3) is valid as the total number of digits are increased by 2 but scale is increased only by 1 which will not lead to any data loss.
- The allowed range is 38,38 (precision, scale) and is a valid upper case scenario which is not resulting in data loss.

Examples

- Changing data type of column a1 from INT to BIGINT.

  ALTER TABLE *test_db.carbon* **CHANGE** *a1 a1 BIGINT*;
- Changing decimal precision of column a1 from 10 to 18.

  ALTER TABLE *test_db.carbon* **CHANGE** *a1 a1 DECIMAL(18,2)*;

System Response

By running DESCRIBE command, the changed data type for the modified column is displayed.

12.3.6.1.10 REFRESH TABLE

Function

This command is used to register Carbon table to Hive meta store catalogue from exisiting Carbon table data.

Syntax

REFRESH TABLE *db_name.table_name*;

Parameter Description

Table 12-34 REFRESH TABLE parameters

Parameter	Description
db_name	Name of the database. If this parameter is left unspecified, the current database is selected.
table_name	Name of the table.

Usage Guidelines

- The new database name and the old database name should be same.
- Before executing this command the old table schema and data should be copied into the new database location.
- If the table is aggregate table, then all the aggregate tables should be copied to the new database location.
- For old store, the time zone of the source and destination cluster should be same.
- If old cluster used HIVE meta store to store schema, refresh will not work as schema file does not exist in file system.

Examples

REFRESH TABLE *dbcarbon.productSalesTable*;

System Response

By running this command, the Carbon table will be registered to Hive meta store catalogue from exisiting Carbon table data.

12.3.6.1.11 REGISTER INDEX TABLE

Function

This command is used to register an index table with the primary table.

Syntax

REGISTER INDEX TABLE *indextable_name* ON *db_name.maintable_name*;

Parameter Description

Table 12-35 REFRESH INDEX TABLE parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is selected.
indextable_name	Index table name.
maintable_name	Primary table name.

Precautions

Before running this command, run **REFRESH TABLE** to register the primary table and secondary index table with the Hive metastore.

Examples

create database productdb;

use productdb;

CREATE TABLE productSalesTable(a int,b string,c string) stored as carbondata;

create index productNameIndexTable on table productSalesTable(c) as 'carbondata';

insert into table productSalesTable select 1,'a','aaa';

create database productdb2;

Run the **hdfs** command to copy **productSalesTable** and **productNameIndexTable** in the **productdb** database to the **productdb2** database.

refresh table productdb2.productSalesTable ;

refresh table productdb2.productNameIndexTable ;

explain select * from productdb2.productSalesTable where c = 'aaa'; / The query command does not use an index table.

REGISTER INDEX TABLE productNameIndexTable ON productdb2.productSalesTable;

explain select * from productdb2.productSalesTable where c = 'aaa'; // The query command uses an index table.

System Response

By running this command, the index table will be registered to the primary table.

12.3.6.2 DML

12.3.6.2.1 LOAD DATA

Function

This command is used to load user data of a particular type, so that CarbonData can provide good query performance.

NOTE

Only the raw data on HDFS can be loaded.

Syntax

LOAD DATA *INPATH 'folder_path' INTO TABLE [db_name.]table_name OPTIONS(property_name=property_value, ...);*

Parameter Description

Table 12-36 LOAD DATA parameters

Parameter	Description
folder_path	Path of the file or folder used for storing the raw CSV data.
db_name	Database name. If this parameter is not specified, the current database is used.
table_name	Name of a table in a database.

Precautions

The following configuration items are involved during data loading:

- **DELIMITER**: Delimiters and quote characters provided in the load command. The default value is a comma (**,**).

  OPTIONS('DELIMITER'=',' , 'QUOTECHAR'='"')

  You can use **'DELIMITER'='\t'** to separate CSV data using tabs.

  OPTIONS('DELIMITER'='\t')

  CarbonData also supports **\001** and **\017** as delimiters.

> NOTE
>
> When the delimiter of CSV data is a single quotation mark ('), the single quotation mark must be enclosed in double quotation marks (" "). For example, 'DELIMITER'= "'".

- **QUOTECHAR**: Delimiters and quote characters provided in the load command. The default value is double quotation marks (**"**).

  OPTIONS('DELIMITER'=',' , 'QUOTECHAR'='"')

- **COMMENTCHAR**: Comment characters provided in the load command. During data loading, if there is a comment character at the beginning of a line, the line is regarded as a comment line and data in the line will not be loaded. The default value is a pound key (#).

  OPTIONS('COMMENTCHAR'='#')

- **FILEHEADER**: If the source file does not contain any header, add a header to the **LOAD DATA** command.

  OPTIONS('FILEHEADER'='column1,column2')

- **ESCAPECHAR**: Is used to perform strict verification of the escape character on CSV files. The default value is backslash (****).

  OPTIONS('ESCAPECHAR'='\')

  > NOTE
  >
  > Enter **ESCAPECHAR** in the CSV data. **ESCAPECHAR** must be enclosed in double quotation marks (" "). For example, "a\b".

- Bad records handling:

  In order for the data processing application to provide benefits, certain data integration is required. In most cases, data quality problems are caused by data sources.

  Methods of handling bad records are as follows:

  – Load all of the data before dealing with the errors.

  – Clean or delete bad records before loading data or stop the loading when bad records are found.

  There are many options for clearing source data during CarbonData data loading, as listed in **Table 12-37**.

Table 12-37 Bad Records Logger

Configuration Item	Default Value	Description
BAD_RECORDS_LOGGER_ENABLE	false	Whether to create logs with details about bad records

Configuration Item	Default Value	Description
BAD_RECORDS_ACTION	FAIL	The four types of actions for bad records are as follows: • **FORCE**: Auto-corrects the data by storing the bad records as NULL. • **REDIRECT**: Bad records are written to the raw CSV instead of being loaded. • **IGNORE**: Bad records are neither loaded nor written to the raw CSV. • **FAIL**: Data loading fails if any bad records are found. **NOTE** In loaded data, if all records are bad records, **BAD_RECORDS_ACTION** is invalid and the load operation fails.
IS_EMPTY_DATA_BAD_RECORD	false	Whether empty data of a column to be considered as bad record or not. If this parameter is set to **false**, empty data ("",', or,) is not considered as bad records. If this parameter is set to **true**, empty data is considered as bad records.
BAD_RECORD_PATH	-	HDFS path where bad records are stored. The default value is **Null**. If bad records logging or bad records operation redirection is enabled, the path must be configured by the user.

Example:

LOAD DATA INPATH *'filepath.csv'* **INTO TABLE** *tablename OPTIONS('BAD_RECORDS_LOGGER_ENABLE'='true', 'BAD_RECORD_PATH'='hdfs://hacluster/tmp/carbon', 'BAD_RECORDS_ACTION'='REDIRECT', 'IS_EMPTY_DATA_BAD_RECORD'='false');*

NOTE

If **REDIRECT** is used, CarbonData will add all bad records into a separate CSV file. However, this file must not be used for subsequent data loading because the content may not exactly match the source record. You must clean up the source record for further data ingestion. This option is used to remind you which records are bad.

- **MAXCOLUMNS**: (Optional) Specifies the maximum number of columns parsed by a CSV parser in a line.

OPTIONS('MAXCOLUMNS'='400')

Table 12-38 MAXCOLUMNS

Name of the Optional Parameter	Default Value	Maximum Value
MAXCOLUMNS	2000	20000

Table 12-39 Behavior chart of MAXCOLUMNS

MAXCOLUMNS Value	Number of Columns in the File Header	Final Value Considered
Not specified in Load options	5	2000
Not specified in Load options	6000	6000
40	7	Max (column count of file header, MAXCOLUMNS value)
22000	40	20000
60	Not specified in Load options	Max (Number of columns in the first line of the CSV file, MAXCOLUMNS value)

NOTE

There must be sufficient executor memory for setting the maximum value of **MAXCOLUMNS Option**. Otherwise, data loading will fail.

- If **SORT_SCOPE** is set to **GLOBAL_SORT** during table creation, you can specify the number of partitions to be used when sorting data. If this parameter is not set or is set to a value less than **1**, the number of map tasks is used as the number of reduce tasks. It is recommended that each reduce task process 512 MB to 1 GB data.

  OPTIONS('GLOBAL_SORT_PARTITIONS'='2')

  NOTE

  To increase the number of partitions, you may need to increase the value of **spark.driver.maxResultSize**, as the sampling data collected in the driver increases with the number of partitions.

- **DATEFORMAT**: Specifies the date format of the table.

  OPTIONS('DATEFORMAT'='dateFormat')

NOTE

Date formats are specified by date pattern strings. The date pattern letters in Carbon are same as in JAVA.

- **TIMESTAMPFORMAT**: Specifies the timestamp of a table.
- *OPTIONS('TIMESTAMPFORMAT'='timestampFormat')*
- **SKIP_EMPTY_LINE**: Ignores empty rows in the CSV file during data loading.

  OPTIONS('SKIP_EMPTY_LINE'='TRUE/FALSE')
- **Optional: SCALE_FACTOR**: Used to control the number of partitions for **RANGE_COLUMN**, **SCALE_FACTOR**. The formula is as follows:

  ```
  splitSize = max(blocklet_size, (block_size - blocklet_size)) * scale_factor
  numPartitions = total size of input data / splitSize
  ```

  The default value is **3**. The value ranges from **1** to **300**.

  OPTIONS('SCALE_FACTOR'='10')

  NOTE

  - If **GLOBAL_SORT_PARTITIONS** and **SCALE_FACTOR** are used at the same time, only **GLOBAL_SORT_PARTITIONS** is valid.
  - The compaction on **RANGE_COLUMN** will use **LOCAL_SORT** by default.

Scenarios

To load a CSV file to a CarbonData table, run the following statement:

LOAD DATA *INPATH 'folder path' INTO TABLE tablename OPTIONS(property_name=property_value, ...);*

Examples

The data in the **data.csv** file is as follows:

```
ID,date,country,name,phonetype,serialname,salary
4,2014-01-21 00:00:00,xxx,aaa4,phone2435,ASD66902,15003
5,2014-01-22 00:00:00,xxx,aaa5,phone2441,ASD90633,15004
6,2014-03-07 00:00:00,xxx,aaa6,phone294,ASD59961,15005
```

CREATE TABLE carbontable(ID int, date Timestamp, country String, name String, phonetype String, serialname String,salary int) STORED AS carbondata;

LOAD DATA *inpath 'hdfs://hacluster/tmp/data.csv' INTO table carbontable*

options('DELIMITER'=',');

System Response

Success or failure will be recorded in the driver logs.

12.3.6.2.2 UPDATE CARBON TABLE

Function

This command is used to update the CarbonData table based on the column expression and optional filtering conditions.

Syntax

- Syntax 1:

  UPDATE <CARBON TABLE> SET (column_name1, column_name2, ... column_name n) = (column1_expression , column2_expression , column3_expression ... column n_expression) [WHERE { <filter_condition> }];

- Syntax 2:

  UPDATE <CARBON TABLE> SET (column_name1, column_name2,) = (select sourceColumn1, sourceColumn2 from sourceTable [WHERE { <filter_condition> }]) [WHERE { <filter_condition> }];

Parameter Description

Table 12-40 UPDATE parameters

Parameter	Description
CARBON TABLE	Name of the CarbonData table to be updated
column_name	Target column to be updated
sourceColumn	Column value of the source table that needs to be updated in the target table
sourceTable	Table from which the records are updated to the target table

Precautions

Note the following before running this command:

- The UPDATE command fails if multiple input rows in the source table are matched with a single row in the target table.
- If the source table generates empty records, the UPDATE operation completes without updating the table.
- If rows in the source table do not match any existing rows in the target table, the UPDATE operation completes without updating the table.
- UPDATE is not allowed in the table with secondary index.
- In a subquery, if the source table and target table are the same, the UPDATE operation fails.
- The UPDATE operation fails if the subquery used in the UPDATE command contains an aggregate function or a GROUP BY clause.

  For example, **update t_carbn01 a set (a.item_type_code, a.profit) = (select b.item_type_cd, sum(b.profit) from t_carbn01b b where item_type_cd =2 group by item_type_code);**.

  In the preceding example, aggregate function **sum(b.profit)** and GROUP BY clause are used in the subquery. As a result, the UPDATE operation will fail.

- If the **carbon.input.segments** property has been set for the queried table, the UPDATE operation fails. To solve this problem, run the following statement before the query:

Syntax:

SET carbon.input.segments. <database_name>. <table_name>=*;

Examples

- Example 1:

  update carbonTable1 d set (d.column3,d.column5) = (select s.c33 ,s.c55 from sourceTable1 s where d.column1 = s.c11) where d.column1 = 'country' exists(select * from table3 o where o.c2 > 1);

- Example 2:

  update carbonTable1 d set (c3) = (select s.c33 from sourceTable1 s where d.column1 = s.c11) where exists(select * from iud.other o where o.c2 > 1);

- Example 3:

  update carbonTable1 set (c2, c5) = (c2 + 1, concat(c5 , "y"));

- Example 4:

  update carbonTable1 d set (c2, c5) = (c2 + 1, "xyx") where d.column1 = 'india';

- Example 5:

  update carbonTable1 d set (c2, c5) = (c2 + 1, "xyx") where d.column1 = 'india' and exists(select * from table3 o where o.column2 > 1);

System Response

Success or failure will be recorded in the driver log and on the client.

12.3.6.2.3 DELETE RECORDS from CARBON TABLE

Function

This command is used to delete records from a CarbonData table.

Syntax

DELETE FROM CARBON_TABLE [WHERE expression];

Parameter Description

Table 12-41 DELETE RECORDS parameters

Parameter	Description
CARBON TABLE	Name of the CarbonData table in which the DELETE operation is performed

Precautions

- If a segment is deleted, all secondary indexes associated with the segment are deleted as well.

- If the **carbon.input.segments** property has been set for the queried table, the DELETE operation fails. To solve this problem, run the following statement before the query:

  Syntax:

  SET carbon.input.segments. <database_name>.<table_name>=*;

Examples

- Example 1:

  delete from columncarbonTable1 d where d.column1 = 'country';

- Example 2:

  delete from dest where column1 IN ('country1', 'country2');

- Example 3:

  delete from columncarbonTable1 where column1 IN (select column11 from sourceTable2);

- Example 4:

  delete from columncarbonTable1 where column1 IN (select column11 from sourceTable2 where column1 = '*');**

- Example 5:

  delete from columncarbonTable1 where column2 >= 4;

System Response

Success or failure will be recorded in the driver log and on the client.

12.3.6.2.4 INSERT INTO CARBON TABLE

Function

This command is used to add the output of the SELECT command to a Carbon table.

Syntax

INSERT INTO [CARBON TABLE] [select query];

Parameter Description

Table 12-42 INSERT INTO parameters

Parameter	Description
CARBON TABLE	Name of the CarbonData table to be inserted
select query	SELECT query on the source table (CarbonData, Hive, and Parquet tables are supported)

Precautions

- A table has been created.
- You must belong to the data loading group in order to perform data loading operations. By default, the data loading group is named **ficommon**.
- CarbonData tables cannot be overwritten.
- The data type of the source table and the target table must be the same. Otherwise, data in the source table will be regarded as bad records.
- The **INSERT INTO** command does not support partial success. If bad records exist, the command fails.
- When you insert data of the source table to the target table, you cannot upload or update data of the source table.

  To enable data loading or updating during the INSERT operation, set the following parameter to **true**.

  carbon.insert.persist.enable=**true**

  By default, the preceding parameters are set to **false**.

  NOTE

  Enabling this property will reduce the performance of the INSERT operation.

Example

create table carbon01(a int,b string,c string) stored as carbondata;

insert into table carbon01 values(1,'a','aa'),(2,'b','bb'),(3,'c','cc');

create table carbon02(a int,b string,c string) stored as carbondata;

INSERT INTO carbon02 select * from carbon01 where a > 1;

System Response

Success or failure will be recorded in the driver logs.

12.3.6.2.5 DELETE SEGMENT by ID

Function

This command is used to delete segments by the ID.

Syntax

DELETE FROM TABLE db_name.table_name WHERE SEGMENT.ID IN (segment_id1,segment_id2);

Parameter Description

Table 12-43 DELETE SEGMENT parameters

Parameter	Description
segment_id	ID of the segment to be deleted.
db_name	Database name. If the parameter is not specified, the current database is used.
table_name	The name of the table in a specific database.

Usage Guidelines

Segments cannot be deleted from the stream table.

Examples

DELETE FROM TABLE CarbonDatabase.CarbonTable WHERE SEGMENT.ID IN (0);

DELETE FROM TABLE CarbonDatabase.CarbonTable WHERE SEGMENT.ID IN (0,5,8);

System Response

Success or failure will be recorded in the CarbonData log.

12.3.6.2.6 DELETE SEGMENT by DATE

Function

This command is used to delete segments by loading date. Segments created before a specific date will be deleted.

Syntax

DELETE FROM TABLE db_name.table_name WHERE SEGMENT.STARTTIME BEFORE date_value;

Parameter Description

Table 12-44 DELETE SEGMENT by DATE parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is used.
table_name	Name of a table in the specified database

Parameter	Description
date_value	Valid date when segments are started to be loaded. Segments before the date will be deleted.

Precautions

Segments cannot be deleted from the stream table.

Example

DELETE FROM TABLE db_name.table_name WHERE SEGMENT.STARTTIME BEFORE '2017-07-01 12:07:20';

STARTTIME indicates the loading start time of different loads.

System Response

Success or failure will be recorded in CarbonData logs.

12.3.6.2.7 SHOW SEGMENTS

Function

This command is used to list the segments of a CarbonData table.

Syntax

SHOW SEGMENTS FOR TABLE *[db_name.]table_name* **LIMIT** *number_of_loads;*

Parameter Description

Table 12-45 SHOW SEGMENTS FOR TABLE parameters

Parameter	Description
db_name	Database name. If this parameter is not specified, the current database is used.
table_name	Name of a table in the specified database
number_of_loads	Threshold of records to be listed

Precautions

None

Examples

create table carbon01(a int,b string,c string) stored as carbondata;

insert into table carbon01 select 1,'a','aa';

insert into table carbon01 select 2,'b','bb';

insert into table carbon01 select 3,'c','cc';

SHOW SEGMENTS FOR TABLE carbon01 LIMIT 2;

System Response

```
+-----+----------+--------------------------+------------------+------------+------------+-------------+--------------+--
+
| ID  |  Status  |     Load Start Time      | Load Time Taken  | Partition  | Data Size  | Index Size  | File Format  |
+-----+----------+--------------------------+------------------+------------+------------+-------------+--------------+--
+
| 3   | Success  | 2020-09-28 22:53:26.336  | 3.726S           | {}         | 6.47KB     | 3.30KB      | columnar_v3  |
| 2   | Success  | 2020-09-28 22:53:01.702  | 6.688S           | {}         | 6.47KB     | 3.30KB      | columnar_v3  |
+-----+----------+--------------------------+------------------+------------+------------+-------------+--------------+--
+
```

12.3.6.2.8 CREATE SECONDARY INDEX

Function

This command is used to create secondary indexes in the CarbonData tables.

Syntax

CREATE INDEX *index_name*

ON TABLE *[db_name.]table_name (col_name1, col_name2)*

AS *'carbondata'*

PROPERTIES *('table_blocksize'='256')*;

Parameter Description

Table 12-46 CREATE SECONDARY INDEX parameters

Parameter	Description
index_name	Index table name. It consists of letters, digits, and special characters (_).
db_name	Database name. It consists of letters, digits, and special characters (_).
table_name	Name of the database table. It consists of letters, digits, and special characters (_).
col_name	Name of a column in a table. Multiple columns are supported. It consists of letters, digits, and special characters (_).
table_blocksize	Block size of a data file. For details, see **•Block Size**.

Precautions

db_name is optional.

Examples

create table productdb.productSalesTable(id int,price int,productName string,city string) stored as carbondata;

CREATE INDEX productNameIndexTable on table productdb.productSalesTable (productName,city) as 'carbondata' ;

In this example, a secondary table named **productdb.productNameIndexTable** is created and index information of the provided column is loaded.

System Response

A secondary index table will be created. Index information related to the provided column will be loaded into the secondary index table. The success message will be recorded in system logs.

12.3.6.2.9 SHOW SECONDARY INDEXES

Function

This command is used to list all secondary index tables in the CarbonData table.

Syntax

SHOW INDEXES ON db_name.table_name;

Parameter Description

Table 12-47 SHOW SECONDARY INDEXES parameters

Parameter	Description
db_name	Database name. It consists of letters, digits, and special characters (_).
table_name	Name of the database table. It consists of letters, digits, and special characters (_).

Precautions

db_name is optional.

Examples

create table productdb.productSalesTable(id int,price int,productName string,city string) stored as carbondata;

CREATE INDEX productNameIndexTable on table productdb.productSalesTable (productName,city) as 'carbondata' ;

SHOW INDEXES ON productdb.productSalesTable;

System Response

All index tables and corresponding index columns in a given CarbonData table will be listed.

12.3.6.2.10 DROP SECONDARY INDEX

Function

This command is used to delete the existing secondary index table in a specific table.

Syntax

DROP INDEX *[IF EXISTS] index_name* **ON** *[db_name.]table_name*;

Parameter Description

Table 12-48 DROP SECONDARY INDEX parameters

Parameter	Description
index_name	Name of the index table. Table name contains letters, digits, and underscores (_).
db_Name	Name of the database. If the parameter is not specified, the current database is used.
table_name	Name of the table to be deleted.

Usage Guidelines

In this command, **IF EXISTS** and **db_name** are optional.

Examples

DROP INDEX *if exists productNameIndexTable* **ON** *productdb.productSalesTable*;

System Response

Secondary Index Table will be deleted. Index information will be cleared in CarbonData table and the success message will be recorded in system logs.

12.3.6.2.11 CLEAN FILES

Function

After the **DELETE SEGMENT** command is executed, the deleted segments are marked as the **delete** state. After the segments are merged, the status of the original segments changes to **compacted**. The data files of these segments are not physically deleted. If you want to forcibly delete these files, run the **CLEAN FILES** command.

However, running this command may result in a query command execution failure.

Syntax

CLEAN FILES FOR TABLE *[db_name.]table_name* ;

Parameter Description

Table 12-49 CLEAN FILES FOR TABLE parameters

Parameter	Description
db_name	Database name. It consists of letters, digits, and underscores (_).
table_name	Name of the database table. It consists of letters, digits, and underscores (_).

Precautions

None

Examples

Add Carbon configuration parameters.

```
carbon.clean.file.force.allowed = true
```

create table carbon01(a int,b string,c string) stored as carbondata;

insert into table carbon01 select 1,'a','aa';

insert into table carbon01 select 2,'b','bb';

delete from table carbon01 where segment.id in (0);

show segments for table carbon01;

CLEAN FILES FOR TABLE carbon01 options('force'='true');

show segments for table carbon01;

In this example, all the segments marked as **deleted** and **compacted** are physically deleted.

System Response

Success or failure will be recorded in the driver logs.

12.3.6.2.12 SET/RESET

Function

This command is used to dynamically add, update, display, or reset the CarbonData properties without restarting the driver.

Syntax

- Add or Update parameter value:

  SET *parameter_name=parameter_value*

  This command is used to add or update the value of **parameter_name**.
- Display property value:

  SET *parameter_name*

  This command is used to display the value of **parameter_name**.
- Display session parameter:

  SET

  This command is used to display all supported session parameters.
- Display session parameters along with usage details:

  SET -v

  This command is used to display all supported session parameters and their usage details.
- Reset parameter value:

  RESET

  This command is used to clear all session parameters.

Parameter Description

Table 12-50 SET parameters

Parameter	Description
parameter_name	Name of the parameter whose value needs to be dynamically added, updated, or displayed
parameter_value	New value of **parameter_name** to be set

Precautions

The following table lists the properties which you can set or clear using the SET or RESET command.

Table 12-51 Properties

Property	Description
carbon.options.bad.records.logger.enable	Whether to enable bad record logger.
carbon.options.bad.records.action	Operations on bad records, for example, force, redirect, fail, or ignore. For more information, see **•Bad record handling**.
carbon.options.is.empty.data.bad.record	Whether the empty data is considered as a bad record. For more information, see **Bad record handling**.
carbon.options.sort.scope	Scope of the sort during data loading.
carbon.options.bad.record.path	HDFS path where bad records are stored.
carbon.custom.block.distribution	Whether to enable Spark or CarbonData block distribution.
enable.unsafe.sort	Whether to use unsafe sort during data loading. Unsafe sort reduces the garbage collection during data loading, thereby achieving better performance.
carbon.si.lookup.partialstring	If this is set to **TRUE**, the secondary index uses the starts-with, ends-with, contains, and LIKE partition condition strings. If this is set to **FALSE**, the secondary index uses only the starts-with partition condition string.

Property	Description
carbon.input.segments	Segment ID to be queried. This property allows you to query a specified segment of a specified table. CarbonScan reads data only from the specified segment ID. Syntax: **carbon.input.segments.<database_name>.<table_name> = <list of segment ids >** If you want to query a specified segment in multi-thread mode, you can use **CarbonSession.threadSet** instead of the **SET** statement. Syntax: **CarbonSession.threadSet ("carbon.input.segments.<database_name>.<table_name>","<list of segment ids >");** **NOTE** You are advised not to set this property in the **carbon.properties** file because all sessions contain the segment list unless session-level or thread-level overwriting occurs.

Examples

- Add or Update:

  SET *enable.unsafe.sort=true*

- Display property value:

  SET *enable.unsafe.sort*

- Show the segment ID list, segment status, and other required details, and specify the segment list to be read:

  SHOW SEGMENTS FOR *TABLE carbontable1;*

  SET *carbon.input.segments.db.carbontable1 = 1, 3, 9;*

- Query a specified segment in multi-thread mode:

  CarbonSession.threadSet (*"carbon.input.segments.default.carbon_table_MulTI_THread", "1,3"*);

- Use **CarbonSession.threadSet** to query segments in a multi-thread environment (Scala code is used as an example):

```
def main(args: Array[String]) {
 Future
{             CarbonSession.threadSet("carbon.input.segments.default.carbon_table_MulTI_THread", "1")
    spark.sql("select count(empno) from carbon_table_MulTI_THread").show()
  }
}
```

- Reset:
  RESET

System Response

- Success will be recorded in the driver log.
- Failure will be displayed on the UI.

12.3.6.3 Operation Concurrent Execution

Before performing **DDL** and **DML** operations, you need to obtain the corresponding locks. See **Table 12-52** for details about the locks that need to be obtained for each operation. The check mark (√) indicates that the lock is required. An operation can be performed only after all required locks are obtained.

You can check whether any two operations can be executed concurrently by using the following method: The first two lines in **Table 12-52** indicate two operations. If no column in the two lines is marked with the check mark (√), the two operations can be executed concurrently. That is, if the columns with check marks (√) in the two lines do not exist, the two operations can be executed concurrently.

Table 12-52 List of obtaining locks for operations

Operation	METADATA_LOCK	COMPACTION_LOCK	DROP_TABLE_LOCK	DELETE_SEGMENT_LOCK	CLEAN_FILES_LOCK	ALTER_PARTITION_LOCK	UPDATE_LOCK	STREAMING_LOCK	CONCURRENT_LOAD_LOCK	SEGMENT_LOCK
CREATE TABLE	-	-	-	-	-	-	-	-	-	-
CREATE TABLE As SELECT	-	-	-	-	-	-	-	-	-	-
DROP TABLE	√	-	√	-	-	-	-	√	-	-
ALTER TABLE COMPACTION	-	√	-	-	-	-	√	-	-	-

Operation	METADATA_LOCK	COMPACTION_LOCK	DROP_TABLE_LOCK	DELETE_SEGMENT_LOCK	CLEAN_FILES_LOCK	ALTER_PARTITION_LOCK	UPDATE_LOCK	STREAMING_LOCK	CONCURRENT_LOAD_LOCK	SEGMENT_LOCK
TABLE RENAME	-	-	-	-	-	-	-	-	-	-
ADD COLUMNS	√	√	-	-	-	-	-	-	-	-
DROP COLUMNS	√	√	-	-	-	-	-	-	-	-
CHANGE DATA TYPE	√	√	-	-	-	-	-	-	-	-
REFRESH TABLE	-	-	-	-	-	-	-	-	-	-
REGISTER INDEX TABLE	√	-	-	-	-	-	-	-	-	-
REFRESH INDEX	-	√	-	-	-	-	-	-	-	-
LOAD DATA/ INSERT INTO	-	-	-	-	-	-	-	-	√	√

Operation	METADATA_LOCK	COMPACTION_LOCK	DROP_TABLE_LOCK	DELETE_SEGMENT_LOCK	CLEAN_FILES_LOCK	ALTER_PARTITION_LOCK	UPDATE_LOCK	STREAMING_LOCK	CONCURRENT_LOAD_LOCK	SEGMENT_LOCK
UPDATE CARBON TABLE	√	√	-	-	-	-	√	-	-	-
DELETE RECORDS from CARBON TABLE	√	√	-	-	-	-	√	-	-	-
DELETE SEGMENT by ID	-	-	-	√	√	-	-	-	-	-
DELETE SEGMENT by DATE	-	-	-	√	√	-	-	-	-	-
SHOW SEGMENTS	-	-	-	-	-	-	-	-	-	-
CREATE SECONDARY INDEX	√	√	-	√	-	-	-	-	-	-

Operation	METADATA_LOCK	COMPACTION_LOCK	DROP_TABLE_LOCK	DELETE_SEGMENT_LOCK	CLEAN_FILES_LOCK	ALTER_PARTITION_LOCK	UPDATE_LOCK	STREAMING_LOCK	CONCURRENT_LOAD_LOCK	SEGMENT_LOCK
SHOW SECONDARY INDEXES	-	-	-	-	-	-	-	-	-	-
DROP SECONDARY INDEX	√	-	√	-	-	-	-	-	-	-
CLEAN FILES	-	-	-	-	-	-	-	-	-	-
SET/RESET	-	-	-	-	-	-	-	-	-	-
Add Hive Partition	-	-	-	-	-	-	-	-	-	-
Drop Hive Partition	√	√	√	√	√	√	-	-	-	-
Drop Partition	√	√	√	√	√	√	-	-	-	-
Alter table set	√	√	-	-	-	-	-	-	-	-

12.3.6.4 API

This section describes the APIs and usage methods of Segment. All methods are in the org.apache.spark.util.CarbonSegmentUtil class.

The following methods have been abandoned:

```
/**
* Returns the valid segments for the query based on the filter condition
* present in carbonScanRdd.
*
* @param carbonScanRdd
* @return Array of valid segments
*/
@deprecated def getFilteredSegments(carbonScanRdd: CarbonScanRDD[InternalRow]): Array[String];
```

Usage Method

Use the following methods to obtain CarbonScanRDD from the query statement:

```
val df=carbon.sql("select * from table where age='12'")
val myscan=df.queryExecution.sparkPlan.collect {
case scan: CarbonDataSourceScan if scan.rdd.isInstanceOf[CarbonScanRDD[InternalRow]] => scan.rdd
case scan: RowDataSourceScanExec if scan.rdd.isInstanceOf[CarbonScanRDD[InternalRow]] => scan.rdd
}.head
val carbonrdd=myscan.asInstanceOf[CarbonScanRDD[InternalRow]]
```

Example:

```
CarbonSegmentUtil.getFilteredSegments(carbonrdd)
```

The filtered segment can be obtained by importing SQL statements.

```
/**
* Returns an array of valid segment numbers based on the filter condition provided in the sql
* NOTE: This API is supported only for SELECT Sql (insert into,ctas,.., is not supported)
*
* @param sql
* @param sparkSession
* @return Array of valid segments
* @throws UnsupportedOperationException because Get Filter Segments API supports if and only
* if only one carbon main table is present in query.
*/
def getFilteredSegments(sql: String, sparkSession: SparkSession): Array[String];
```

Example:

```
CarbonSegmentUtil.getFilteredSegments("select * from table where age='12'", sparkSession)
```

Import the database name and table name to obtain the list of segments to be merged. The obtained segments can be used as parameters of the getMergedLoadName function.

```
/**
* Identifies all segments which can be merged with MAJOR compaction type.
* NOTE: This result can be passed to getMergedLoadName API to get the merged load name.
*
* @param sparkSession
* @param tableName
* @param dbName
* @return list of LoadMetadataDetails
*/
def identifySegmentsToBeMerged(sparkSession: SparkSession,
tableName: String,
dbName: String) : util.List[LoadMetadataDetails];
```

Example:

```
CarbonSegmentUtil.identifySegmentsToBeMerged(sparkSession, "table_test","default")
```

Import the database name, table name, and obtain all segments which can be merged with CUSTOM compaction type. The obtained segments can be transferred as the parameter of the getMergedLoadName function.

```
/**
* Identifies all segments which can be merged with CUSTOM compaction type.
*  NOTE: This result can be passed to getMergedLoadName API to get the merged load name.
*
* @param sparkSession
* @param tableName
* @param dbName
* @param customSegments
* @return list of LoadMetadataDetails
* @throws UnsupportedOperationException if customSegments is null or empty.
* @throws MalformedCarbonCommandException if segment does not exist or is not valid
*/
def identifySegmentsToBeMergedCustom(sparkSession: SparkSession,
tableName: String,
dbName: String,
customSegments: util.List[String]): util.List[LoadMetadataDetails];
```

Example:

```
val customSegments = new util.ArrayList[String]()
customSegments.add("1")
customSegments.add("2")
CarbonSegmentUtil.identifySegmentsToBeMergedCustom(sparkSession, "table_test","default",
customSegments)
```

If a segment list is specified, the merged load name is returned.

```
/**
* Returns the Merged Load Name for given list of segments
*
* @param list of segments
* @return Merged Load Name
* @throws UnsupportedOperationException if list of segments is less than 1
*/
def getMergedLoadName(list: util.List[LoadMetadataDetails]): String;
```

Example:

```
val carbonTable = CarbonEnv.getCarbonTable(Option(databaseName), tableName)(sparkSession)
val loadMetadataDetails = SegmentStatusManager.readLoadMetadata(carbonTable.getMetadataPath)
CarbonSegmentUtil.getMergedLoadName(loadMetadataDetails.toList.asJava)
```

12.3.6.5 Spatial Indexes

Quick Example

```
create table IF NOT EXISTS carbonTable
(
COLUMN1    BIGINT,
LONGITUDE    BIGINT,
LATITUDE    BIGINT,
COLUMN2    BIGINT,
COLUMN3    BIGINT
)
STORED AS carbondata
TBLPROPERTIES
('SPATIAL_INDEX.mygeohash.type'='geohash','SPATIAL_INDEX.mygeohash.sourcecolumns'='longitude,
latitude','SPATIAL_INDEX.mygeohash.originLatitude'='39.850713','SPATIAL_INDEX.mygeohash.gridSize'='50','S
PATIAL_INDEX.mygeohash.minLongitude'='115.828503','SPATIAL_INDEX.mygeohash.maxLongitude'='720.000
000','SPATIAL_INDEX.mygeohash.minLatitude'='39.850713','SPATIAL_INDEX.mygeohash.maxLatitude'='720.0
00000','SPATIAL_INDEX'='mygeohash','SPATIAL_INDEX.mygeohash.conversionRatio'='1000000','SORT_COLU
MNS'='column1,column2,column3,latitude,longitude');
```

Introduction to Spatial Indexes

Spatial data includes multidimensional points, lines, rectangles, cubes, polygons, and other geometric objects. A spatial data object occupies a certain region of

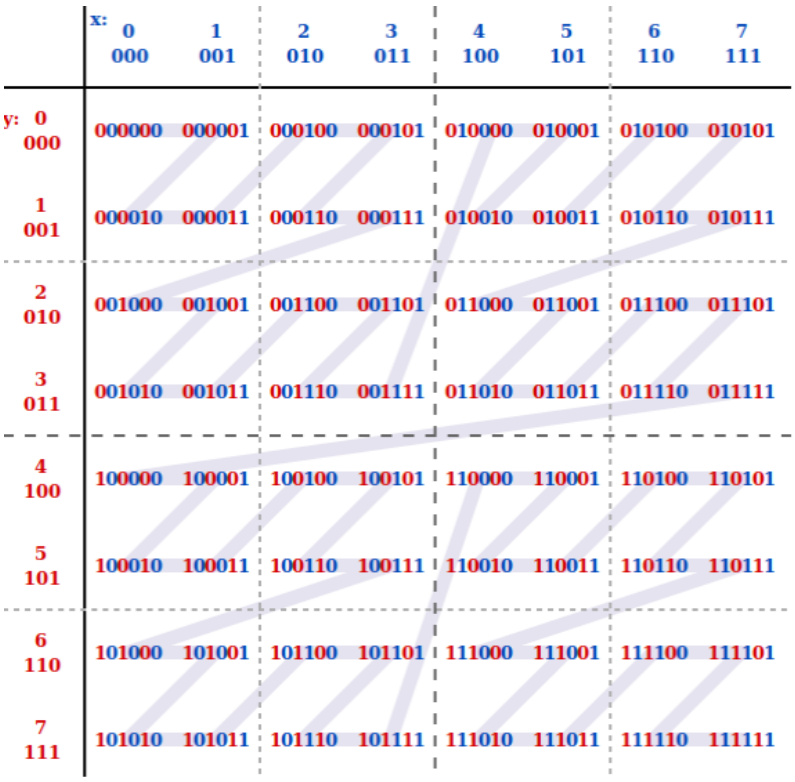
space, called spatial scope, characterized by its location and boundary. The spatial data can be either point data or region data.

- Point data: A point has a spatial extent characterized completely by its location. It does not occupy space and has no associated boundary. Point data consists of a collection of points in a two-dimensional space. Points can be stored as a pair of longitude and latitude.
- Region data: A region has a spatial extent with a location, and boundary. The location can be considered as the position of a fixed point in the region, such as its centroid. In two dimensions, the boundary can be visualized as a line (for finite regions, a closed loop). Region data contains a collection of regions.

Currently, only point data is supported, and it can be stored.

Longitude and latitude can be encoded as a unique GeoID. Geohash is a public-domain geocoding system invented by Gustavo Niemeyer. It encodes geographical locations into a short string of letters and digits. It is a hierarchical spatial data structure which subdivides the space into buckets of grid shape, which is one of the many applications of what is known as the Z-order curve, and generally the space-filling curve.

The Z value of a point in multiple dimensions is calculated by interleaving the binary representation of its coordinate value, as shown in the following figure. When Geohash is used to create a GeoID, data is sorted by GeoID instead of longitude and latitude. Data is stored by spatial proximity.



Creating a Table

GeoHash encoding:

```
create table IF NOT EXISTS carbonTable
(
```

```
...
`LONGITUDE`     BIGINT,
`LATITUDE`      BIGINT,
...
)
STORED AS carbondata
TBLPROPERTIES
('SPATIAL_INDEX.mygeohash.type'='geohash','SPATIAL_INDEX.mygeohash.sourcecolumns'='longitude,
latitude','SPATIAL_INDEX.mygeohash.originLatitude'='xx.xxxxxx','SPATIAL_INDEX.mygeohash.gridSize'='xx','SP
ATIAL_INDEX.mygeohash.minLongitude'='xxx.xxxxxx','SPATIAL_INDEX.mygeohash.maxLongitude'='xxx.xxxxxx'
,'SPATIAL_INDEX.mygeohash.minLatitude'='xx.xxxxxx','SPATIAL_INDEX.mygeohash.maxLatitude'='xxx.xxxxxx','
SPATIAL_INDEX'='mygeohash','SPATIAL_INDEX.mygeohash.conversionRatio'='1000000','SORT_COLUMNS'='co
lumn1,column2,column3,latitude,longitude');
```

SPATIAL_INDEX is a user-defined index handler. This handler allows users to create new columns from the table-structure column set. The new column name is the same as that of the handler name. The **type** and **sourcecolumns** properties of the handler are mandatory. Currently, the value of **type** supports only **geohash**. Carbon provides a default implementation class that can be easily used. You can extend the default implementation class to mount the customized implementation class of **geohash**. The default handler also needs to provide the following table properties:

- **SPATIAL_INDEX.*xxx*.originLatitude**: specifies the origin latitude. (**Double** type.)
- **SPATIAL_INDEX.*xxx*.gridSize**: specifies the grid length in meters. (**Int** type.)
- **SPATIAL_INDEX.*xxx*.minLongitude**: specifies the minimum longitude. (**Double** type.)
- **SPATIAL_INDEX.*xxx*.maxLongitude**: specifies the maximum longitude. (**Double** type.)
- **SPATIAL_INDEX.*xxx*.minLatitude**: specifies the minimum latitude. (**Double** type.)
- **SPATIAL_INDEX.*xxx*.maxLatitude**: specifies the maximum latitude. (**Double** type.)
- **SPATIAL_INDEX.*xxx*.conversionRatio**: used to convert the small value of the longitude and latitude to an integer. (**Int** type.)

You can add your own table properties to the handlers in the above format and access them in your custom implementation class. **originLatitude**, **gridSize**, and **conversionRatio** are mandatory. Other parameters are optional in Carbon. You can use the **SPATIAL_INDEX.*xxx*.class** property to specify their implementation classes.

The default implementation class can generate handler column values for **sourcecolumns** in each row and support query based on the **sourcecolumns** filter criteria. The generated handler column is invisible to users. Except the **SORT_COLUMNS** table properties, no DDL commands or properties are allowed to contain the handler column.

NOTE

- By default, the generated handler column is regarded as the sorting column. If **SORT_COLUMNS** does not contain any **sourcecolumns**, add the handler column to the end of the existing **SORT_COLUMNS**. If the handler column has been specified in **SORT_COLUMNS**, its order in **SORT_COLUMNS** remains unchanged.
- If **SORT_COLUMNS** contains any **sourcecolumns** but does not contain the handler column, the handler column is automatically inserted before **sourcecolumns** in **SORT_COLUMNS**.
- If **SORT_COLUMNS** needs to contain any **sourcecolumns**, ensure that the handler column is listed before the **sourcecolumns** so that the handler column can take effect during sorting.

GeoSOT encoding:

```
CREATE TABLE carbontable(
...
longitude DOUBLE,
latitude DOUBLE,
...)
STORED AS carbondata
TBLPROPERTIES ('SPATIAL_INDEX'='xxx',
'SPATIAL_INDEX.xxx.type'='geosot',
'SPATIAL_INDEX.xxx.sourcecolumns'='longitude, latitude',
'SPATIAL_INDEX.xxx.level'='21',
'SPATIAL_INDEX.xxx.class'='org.apache.carbondata.geo.GeoSOTIndex')
```

Table 12-53 Parameter description

Parameter	Description
SPATIAL_INDEX	Specifies the spatial index. Its value is the same as the column name.
SPATIAL_INDEX.xxx.type	(Mandatory) The value is set to **geosot**.
SPATIAL_INDEX.xxx.source columns	(Mandatory) Specifies the source columns for calculating the spatial index. The value must be two existing columns of the double type.
SPATIAL_INDEX.xxx.level	(Optional) Specifies the columns for calculating the spatial index. The default value is **17**, through which you can obtain an accurate result and improve the computing performance.
SPATIAL_INDEX.xxx.class	(Optional) Specifies the implementation class of GeoSOT. The default value is **org.apache.carbondata.geo.GeoSOTIndex**.

Example:

```
create table geosot(
timevalue bigint,
longitude double,
latitude double)
stored as carbondata
TBLPROPERTIES ('SPATIAL_INDEX'='mygeosot',
'SPATIAL_INDEX.mygeosot.type'='geosot',
'SPATIAL_INDEX.mygeosot.level'='21', 'SPATIAL_INDEX.mygeosot.sourcecolumns'='longitude, latitude');
```

Preparing Data

- Data file 1: **geosotdata.csv**

```
timevalue,longitude,latitude
1575428400000,116.285807,40.084087
1575428400000,116.372142,40.129503
1575428400000,116.187332,39.979316
1575428400000,116.337069,39.951887
1575428400000,116.359102,40.154684
1575428400000,116.736367,39.970323
1575428400000,116.720179,40.009893
1575428400000,116.346961,40.13355
1575428400000,116.302895,39.930753
1575428400000,116.288955,39.999101
1575428400000,116.17609,40.129953
1575428400000,116.725575,39.981115
1575428400000,116.266922,40.179415
1575428400000,116.353706,40.156483
1575428400000,116.362699,39.942444
1575428400000,116.325378,39.963129
```

- Data file 2: **geosotdata2.csv**

```
timevalue,longitude,latitude
1575428400000,120.17708,30.326882
1575428400000,120.180685,30.326327
1575428400000,120.184976,30.327105
1575428400000,120.189311,30.327549
1575428400000,120.19446,30.329698
1575428400000,120.186965,30.329133
1575428400000,120.177481,30.328911
1575428400000,120.169713,30.325614
1575428400000,120.164563,30.322243
1575428400000,120.171558,30.319613
1575428400000,120.176365,30.320687
1575428400000,120.179669,30.323688
1575428400000,120.181001,30.320761
1575428400000,120.187094,30.32354
1575428400000,120.193574,30.323651
1575428400000,120.186192,30.320132
1575428400000,120.190055,30.317464
1575428400000,120.195376,30.318094
1575428400000,120.160786,30.317094
1575428400000,120.168211,30.318057
1575428400000,120.173618,30.316612
1575428400000,120.181001,30.317316
1575428400000,120.185162,30.315908
1575428400000,120.192415,30.315871
1575428400000,120.161902,30.325614
1575428400000,120.164306,30.328096
1575428400000,120.197093,30.325985
1575428400000,120.19602,30.321651
1575428400000,120.198638,30.32354
1575428400000,120.165421,30.314834
```

Importing Data

The GeoHash default implementation class extends the customized index abstract class. If the handler property is not set to a customized implementation class, the default implementation class is used. You can extend the default implementation class to mount the customized implementation class of **geohash**. The methods of the customized index abstract class are as follows:

- **Init** method: Used to extract, verify, and store the handler property. If the operation fails, the system throws an exception and displays the error information.

- **Generate** method: Used to generate indexes. It generates an index for each row of data.
- **Query** method: Used to generate an index value range list for given input.

The commands for importing data are the same as those for importing common Carbon tables.

LOAD DATA inpath '/tmp/*geosotdata.csv*' INTO TABLE geosot OPTIONS ('DELIMITER'= ',');

LOAD DATA inpath '/tmp/*geosotdata2.csv*' INTO TABLE geosot OPTIONS ('DELIMITER'= ',');

NOTE

For details about **geosotdata.csv** and **geosotdata2.csv**, see **Preparing Data**.

Aggregate Query of Irregular Spatial Sets

Query statements and filter UDFs

- Filtering data based on polygon

  IN_POLYGON(pointList)

  UDF input parameter

Parameter	Type	Description
pointList	String	Enter multiple points as a string. Each point is presented as **longitude latitude**. Longitude and latitude are separated by a space. Each pair of longitude and latitude is separated by a comma (,). The longitude and latitude values at the start and end of the string must be the same.

  UDF output parameter

Parameter	Type	Description
inOrNot	Boolean	Checks whether data is in the specified **polygon_list**.

  Example:

```
select longitude, latitude from geosot where IN_POLYGON('116.321011 40.123503, 116.137676 39.947911, 116.560993 39.935276, 116.321011 40.123503');
```

- Filtering data based on the polygon list

  IN_POLYGON_LIST(polygonList, opType)

  UDF input parameters

Parameter	Type	Description
polygonList	String	Inputs multiple polygons as a string. Each polygon is presented as **POLYGON ((longitude1 latitude1, longitude2 latitude2, ...))**. Note that there is a space after **POLYGON**. Longitudes and latitudes are separated by spaces. Each pair of longitude and latitude is separated by a comma (,). The longitudes and latitudes at the start and end of a polygon must be the same. **IN_POLYGON_LIST** requires at least two polygons. Example: POLYGON ((116.137676 40.163503, 116.137676 39.935276, 116.560993 39.935276, 116.137676 40.163503))
opType	String	Performs union, intersection, and subtraction on multiple polygons. Currently, the following operation types are supported: ● OR: A U B U C (Assume that three polygons A, B, and C are input.) ● AND: A ∩ B ∩ C

UDF output parameter

Parameter	Type	Description
inOrNot	Boolean	Checks whether data is in the specified **polygon_list**.

Example:

```
select longitude, latitude from geosot where IN_POLYGON_LIST('POLYGON ((120.176433 30.327431,120.171283 30.322245,120.181411 30.314540, 120.190509 30.321653,120.185188 30.329358,120.176433 30.327431)), POLYGON ((120.191603 30.328946,120.184179 30.327465,120.181819 30.321464, 120.190359 30.315388,120.199242 30.324464,120.191603 30.328946))', 'OR');
```

- Filtering data based on the polyline list

  IN_POLYLINE_LIST(polylineList, bufferInMeter)

  UDF input parameters

Parameter	Type	Description
polylineList	String	Inputs multiple polylines as a string. Each polyline is presented as **LINESTRING (longitude1 latitude1, longitude2 latitude2, ...)**. Note that there is a space after **LINESTRING**. Longitudes and latitudes are separated by spaces. Each pair of longitude and latitude is separated by a comma (,). A union will be output based on the data in multiple polylines. Example: `LINESTRING (116.137676 40.163503, 116.137676 39.935276, 116.260993 39.935276)`
bufferInMeter	Float	Polyline buffer distance, in meters. Right angles are used at the end to create a buffer.

UDF output parameter

Parameter	Type	Description
inOrNot	Boolean	Checks whether data is in the specified **polyline_list**.

Example:

```
select longitude, latitude from geosot where IN_POLYLINE_LIST('LINESTRING (120.184179 30.327465, 120.191603 30.328946, 120.199242 30.324464, 120.190359 30.315388)', 65);
```

- Filtering data based on the GeoID range list

  IN_POLYGON_RANGE_LIST(polygonRangeList, opType)

  UDF input parameters

Parameter	Type	Description
polygonRangeList	String	Inputs multiple rangeLists as a string. Each rangeList is presented as **RANGELIST (startGeoId1 endGeoId1, startGeoId2 endGeoId2, ...)**. Note that there is a space after **RANGELIST**. Start GeoIDs and end GeoIDs are separated by spaces. Each group of GeoID ranges is separated by a comma (,). Example: RANGELIST (855279368848 855279368850, 855280799610 855280799612, 855282156300 855282157400)
opType	String	Performs union, intersection, and subtraction on multiple rangeLists. Currently, the following operation types are supported: ● OR: A U B U C (Assume that three rangeLists A, B, and C are input.) ● AND: A ∩ B ∩ C

UDF output parameter

Parameter	Type	Description
inOrNot	Boolean	Checks whether data is in the specified **polyRange_list**.

Example:

```
select mygeosot, longitude, latitude from geosot where IN_POLYGON_RANGE_LIST('RANGELIST (526549722865860608 526549722865860618, 532555655580483584 532555655580483594)', 'OR');
```

- Performing polygon query

  IN_POLYGON_JOIN(GEO_HASH_INDEX_COLUMN, POLYGON_COLUMN)

  Perform join query on two tables. One is a spatial data table containing the longitude, latitude, and GeoHashIndex columns, and the other is a dimension table that saves polygon data.

  During query, **IN_POLYGON_JOIN UDF**, **GEO_HASH_INDEX_COLUMN**, and **POLYGON_COLUMN** of the polygon table are used. **Polygon_column** specifies the column containing multiple points (longitude and latitude pairs). The first and last points in each row of the Polygon table must be the same. All points in each row form a closed geometric shape.

  UDF input parameters

Parameter	Type	Description
GEO_HASH_INDEX_COLUMN	Long	GeoHashIndex column of the spatial data table.
POLYGON_COLUMN	String	Polygon column of the polygon table, the value of which is represented by the string of polygon, for example, **POLYGON ((longitude1 latitude1, longitude2 latitude2, ...))**.

Example:

```
CREATE TABLE polygonTable(
polygon string,
poiType string,
poiId String)
STORED AS carbondata;

insert into polygonTable select 'POLYGON ((120.176433 30.327431,120.171283 30.322245,
120.181411 30.314540,120.190509 30.321653,120.185188 30.329358,120.176433 30.327431))','abc','1';

insert into polygonTable select 'POLYGON ((120.191603 30.328946,120.184179 30.327465,
120.181819 30.321464,120.190359 30.315388,120.199242 30.324464,120.191603 30.328946))','abc','2';

select t1.longitude,t1.latitude from geosot t1
inner join
(select polygon,poiId from polygonTable where poitype='abc') t2
on in_polygon_join(t1.mygeosot,t2.polygon) group by t1.longitude,t1.latitude;
```

- Performing range_list query

  IN_POLYGON_JOIN_RANGE_LIST(GEO_HASH_INDEX_COLUMN, POLYGON_COLUMN)

  Use the **IN_POLYGON_JOIN_RANGE_LIST** UDF to associate the spatial data table with the polygon dimension table based on **Polygon_RangeList**. By using a range list, you can skip the conversion between a polygon and a range list.

  UDF input parameters

Parameter	Type	Description
GEO_HASH_INDEX_COLUMN	Long	GeoHashIndex column of the spatial data table.
POLYGON_COLUMN	String	Rangelist column of the Polygon table, the value of which is represented by the string of rangeList, for example, **RANGELIST (startGeoId1 endGeoId1, startGeoId2 endGeoId2, ...)**.

Example:

```
CREATE TABLE polygonTable(
polygon string,
```

```
poiType string,
poiId String)
STORED AS carbondata;

insert into polygonTable select 'RANGELIST (526546455897309184 526546455897309284,
526549831217315840 526549831217315850, 532555655580483534 532555655580483584)','xyz','2';

select t1.*
from geosot t1
inner join
(select polygon,poiId from polygonTable where poitype='xyz') t2
on in_polygon_join_range_list(t1.mygeosot,t2.polygon);
```

UDFs of spacial index tools

- Obtaining row number and column number of a grid converted from GeoID

  GeoIdToGridXy(geoId)

  UDF input parameter

Parameter	Type	Description
geoId	Long	Calculates the row number and column number of the grid based on GeoID.

  UDF output parameter

Parameter	Type	Description
gridArray	Array[Int]	Returns the grid row and column numbers contained in GeoID in array. The first digit indicates the row number, and the second digit indicates the column number.

  Example:

```
select longitude, latitude, mygeohash, GeoIdToGridXy(mygeohash) as GridXY from geoTable;
```

- Converting longitude and latitude to GeoID

  LatLngToGeoId(latitude, longitude oriLatitude, gridSize)

  UDF input parameters

Parameter	Type	Description
longitude	Long	Longitude. Note: The value is an integer after conversion.
latitude	Long	Latitude. Note: The value is an integer after conversion.
oriLatitude	Double	Origin latitude, required for calculating GeoID.

Parameter	Type	Description
gridSize	Int	Grid size, required for calculating GeoID.

UDF output parameter

Parameter	Type	Description
geoId	Long	Returns a number that indicates the longitude and latitude after coding.

Example:

```
select longitude, latitude, mygeohash, LatLngToGeoId(latitude, longitude, 39.832277, 50) as geoId from geoTable;
```

- Converting GeoID to longitude and latitude

  GeoIdToLatLng(geoId, oriLatitude, gridSize)

  UDF input parameters

Parameter	Type	Description
geoId	Long	Calculates the longitude and latitude based on GeoID.
oriLatitude	Double	Origin latitude, required for calculating the longitude and latitude.
gridSize	Int	Grid size, required for calculating the longitude and latitude.

  NOTE

  GeoID is generated based on the grid coordinates, which are the grid center. Therefore, the calculated longitude and latitude are the longitude and latitude of the grid center. There may be an error ranging from 0 degree to half of the grid size between the calculated longitude and latitude and the longitude and latitude of the generated GeoID.

  UDF output parameter

Parameter	Type	Description
latitudeAndLongitude	Array[Double]	Returns the longitude and latitude coordinates of the grid center that represent the GeoID in array. The first digit indicates the latitude, and the second digit indicates the longitude.

Example:

```
select longitude, latitude, mygeohash, GeoIdToLatLng(mygeohash, 39.832277, 50) as
LatitudeAndLongitude from geoTable;
```

- Calculating the upper-layer GeoID of the pyramid model

  ToUpperLayerGeoId(geoId)

  UDF input parameter

Parameter	Type	Description
geoId	Long	Calculates the upper-layer GeoID of the pyramid model based on the input GeoID.

  UDF output parameter

Parameter	Type	Description
geoId	Long	Returns the upper-layer GeoID of the pyramid model.

  Example:

```
select longitude, latitude, mygeohash, ToUpperLayerGeoId(mygeohash) as upperLayerGeoId from
geoTable;
```

- Obtaining the GeoID range list using the input polygon

  ToRangeList(polygon, oriLatitude, gridSize)

  UDF input parameters

Parameter	Type	Description
polygon	String	Input polygon string, which is a pair of longitude and latitude. Longitude and latitude are separated by a space. Each pair of longitude and latitude is separated by a comma (,). The longitude and latitude at the start and end must be the same.
oriLatitude	Double	Origin latitude, required for calculating GeoID.
gridSize	Int	Grid size, required for calculating GeoID.

  UDF output parameter

Parameter	Type	Description
geoIdList	Buffer[Array[Long]]	Converts polygons into GeoID range lists.

Example:

```
select ToRangeList('116.321011 40.123503, 116.137676 39.947911, 116.560993 39.935276, 116.321011 40.123503', 39.832277, 50) as rangeList from geoTable;
```

- Calculating the upper-layer longitude of the pyramid model

  ToUpperLongitude (longitude, gridSize, oriLat)

  UDF input parameters

Parameter	Type	Description
longitude	Long	Input longitude, which is a long integer.
gridSize	Int	Grid size, required for calculating longitude.
oriLatitude	Double	Origin latitude, required for calculating longitude.

  UDF output parameter

Parameter	Type	Description
longitude	Long	Returns the upper-layer longitude.

  Example:

```
select ToUpperLongitude (-23575161504L, 50, 39.832277) as upperLongitude from geoTable;
```

- Calculating the upper-layer latitude of the pyramid model

  ToUpperLatitude(Latitude, gridSize, oriLat)

  UDF input parameters

Parameter	Type	Description
latitude	Long	Input latitude, which is a long integer.
gridSize	Int	Grid size, required for calculating latitude.
oriLatitude	Double	Origin latitude, required for calculating latitude.

UDF output parameter

Parameter	Type	Description
Latitude	Long	Returns the upper-layer latitude.

Example:

```
select ToUpperLatitude (-23575161504L, 50, 39.832277) as upperLatitude from geoTable;
```

- Converting longitude and latitude to GeoSOT

  LatLngToGridCode(latitude, longitude, level)

  UDF input parameters

Parameter	Type	Description
latitude	Double	Latitude.
longitude	Double	Longitude.
level	Int	Level. The value range is [0, 32].

UDF output parameter

Parameter	Type	Description
geoId	Long	A number that indicates the longitude and latitude after GeoSOT encoding.

Example:

```
select LatLngToGridCode(39.930753, 116.302895, 21) as geoId;
```

12.3.7 CarbonData Troubleshooting

12.3.7.1 Filter Result Is not Consistent with Hive when a Big Double Type Value Is Used in Filter

Symptom

When double data type values with higher precision are used in filters, incorrect values are returned by filtering results.

Possible Causes

When double data type values with higher precision are used in filters, values are rounded off before comparison. Therefore, values of double data type with different fraction part are considered same.

Troubleshooting Method

NA.

Procedure

To avoid this problem, use decimal data type when high precision data comparisons are required, such as financial applications, equality and inequality checks, and rounding operations.

Reference Information

NA.

12.3.7.2 Query Performance Deterioration

Symptom

The query performance fluctuates when the query is executed in different query periods.

Possible Causes

During data loading, the memory configured for each executor program instance may be insufficient, resulting in more Java GCs. When GC occurs, the query performance deteriorates.

Troubleshooting Method

On the Spark UI, the GC time of some executors is obviously higher than that of other executors, or all executors have high GC time.

Procedure

Log in to Manager and choose **Cluster** > **Services** > **Spark2x**. On the displayed page, click the **Configurations** tab and then **All Configurations**, search for **spark.executor.memory** in the search box, and set its value to a larger value.

Reference

None

12.3.8 CarbonData FAQ

12.3.8.1 Why Is Incorrect Output Displayed When I Perform Query with Filter on Decimal Data Type Values?

Question

Why is incorrect output displayed when I perform query with filter on decimal data type values?

For example:

select * from carbon_table where num = 1234567890123456.22;

Output:

```
+------+---------------------+--+
| name |        num          |
+------+---------------------+--+
| IAA  | 1234567890123456.22 |
| IAA  | 1234567890123456.21 |
+------+---------------------+--+
```

Answer

To obtain accurate output, append BD to the number.

For example:

select * from carbon_table where num = 1234567890123456.22BD;

Output:

```
+------+---------------------+--+
| name |        num          |
+------+---------------------+--+
| IAA  | 1234567890123456.22 |
+------+---------------------+--+
```

12.3.8.2 How to Avoid Minor Compaction for Historical Data?

Question

How to avoid minor compaction for historical data?

Answer

If you want to load historical data first and then the incremental data, perform following steps to avoid minor compaction of historical data:

1. Load all historical data.
2. Configure the major compaction size to a value smaller than the segment size of historical data.
3. Run the major compaction once on historical data so that these segments will not be considered later for minor compaction.
4. Load the incremental data.
5. You can configure the minor compaction threshold as required.

For example:

1. Assume that you have loaded all historical data to CarbonData and the size of each segment is 500 GB.
2. Set the threshold of major compaction property to **carbon.major.compaction.size** = **491520** (480 GB x 1024).
3. Run major compaction. All segments will be compacted because the size of each segment is more than configured size.
4. Perform incremental loading.
5. Configure the minor compaction threshold to **carbon.compaction.level.threshold** = **6,6**.
6. Run minor compaction. As a result, only incremental data is compacted.

12.3.8.3 How to Change the Default Group Name for CarbonData Data Loading?

Question

How to change the default group name for CarbonData data loading?

Answer

By default, the group name for CarbonData data loading is **ficommon**. You can perform the following operation to change the default group name:

1. Edit the **carbon.properties** file.
2. Change the value of the key **carbon.dataload.group.name** as required. The default value is **ficommon**.

12.3.8.4 Why Does INSERT INTO CARBON TABLE Command Fail?

Question

Why does the ***INSERT INTO CARBON TABLE*** command fail and the following error message is displayed?

```
Data load failed due to bad record
```

Answer

The ***INSERT INTO CARBON TABLE*** command fails in the following scenarios:

- If the data type of source and target table columns are not the same, the data from the source table will be treated as bad records and the ***INSERT INTO*** command fails.
- If the result of aggregation function on a source column exceeds the maximum range of the target column, then the ***INSERT INTO*** command fails.

  Solution:

  You can use the cast function on corresponding columns when inserting records.

  For example:

a. Run the ***DESCRIBE*** command to query the target and source table.

DESCRIBE *newcarbontable*;

Result:

```
col1 int
col2 bigint
```

DESCRIBE *sourcetable*;

Result:

```
col1 int
col2 int
```

b. Add the cast function to convert bigint value to integer.

INSERT INTO *newcarbontable select col1, cast(col2 as integer) from sourcetable;*

12.3.8.5 Why Is the Data Logged in Bad Records Different from the Original Input Data with Escape Characters?

Question

Why is the data logged in bad records different from the original input data with escaped characters?

Answer

An escape character is a backslash (\) followed by one or more characters. If the input records contain escape characters such as \t, \b, \n, \r, \f, \', \", \\ , java will process the escape character '\' and the following characters together to obtain the escaped meaning.

For example, if the CSV data type **2010\\10,test** is inserted to String,int type, the value is treated as bad records, because **tes**t cannot be converted to int. The value logged in the bad records is **2010\10** because java processes **** as ****.

12.3.8.6 Why Data Load Performance Decreases due to Bad Records?

Question

Why data load performance decreases due to bad records?

Answer

If bad records are present in the data and **BAD_RECORDS_LOGGER_ENABLE** is **true** or **BAD_RECORDS_ACTION** is **redirect** then load performance will decrease due to extra I/O for writing failure reason in log file or redirecting the records to raw CSV.

12.3.8.7 Why INSERT INTO/LOAD DATA Task Distribution Is Incorrect and the Opened Tasks Are Less Than the Available Executors when the Number of Initial Executors Is Zero?

Question

Why **INSERT INTO or LOAD DATA** task distribution is incorrect, and the openedtasks are less than the available executors when the number of initial executors is zero**?**

Answer

In case of INSERT INTO or LOAD DATA, CarbonData distributes one task per node. If the executors are not allocated from the distinct nodes then CarbonData will launch fewer tasks.

Solution:

Configure higher value for the executor memory and core so that the yarn can launch only one executor per node.

1. Configure the number of the Executor cores.
   - Configure the **spark.executor.cores** in **spark-defaults.conf** or the **SPARK_EXECUTOR_CORES** in **spark-env.sh** appropriately.
   - Add **--executor-cores NUM** parameter to configure the cores during use the spark-submit command.
2. Configure the Executor memory.
   - Configure the **spark.executor.memory** in **spark-defaults.conf** or the **SPARK_EXECUTOR_MEMORY** in **spark-env.sh** appropriately.
   - Add **--executor-memory MEM** parameter to configure the memory during use the spark-submit command.

12.3.8.8 Why Does CarbonData Require Additional Executors Even Though the Parallelism Is Greater Than the Number of Blocks to Be Processed?

Question

Why does CarbonData require additional executors even though the parallelism is greater than the number of blocks to be processed?

Answer

CarbonData block distribution optimizes data processing as follows:

1. Optimize data processing parallelism.
2. Optimize parallel reading of block data.

To optimize parallel processing and parallel read, CarbonData requests executors based on the locality of blocks so that it can obtain executors on all nodes.

If you are using dynamic allocation, you need to configure the following properties:

1. Set **spark.dynamicAllocation.executorIdleTimeout** to 15 minutes (or the average query time).
2. Set **spark.dynamicAllocation.maxExecutors** correctly. The default value **2048** is not recommended. Otherwise, CarbonData will request the maximum number of executors.
3. For a bigger cluster, set **carbon.dynamicAllocation.schedulerTimeout** to a value ranging from 10 to 15 seconds. The default value is 5 seconds.
4. Set **carbon.scheduler.minRegisteredResourcesRatio** to a value ranging from 0.1 to 1.0. The default value is **0.8**. Block distribution can be started as long as the value of **carbon.scheduler.minRegisteredResourcesRatio** is within the range.

12.3.8.9 Why Data loading Fails During off heap?

Question

Why Data Loading fails during off heap?

Answer

YARN Resource Manager will consider (Java heap memory + **spark.yarn.am.memoryOverhead**) as memory limit, so during the off heap, the memory can exceed this limit. So you need to increase the memory by increasing the value of the parameter **spark.yarn.am.memoryOverhead**.

12.3.8.10 Why Do I Fail to Create a Hive Table?

Question

Why do I fail to create a hive table?

Answer

Creating a Hive table fails, when source table or sub query has more number of partitions. The implementation of the query requires a lot of tasks, then the number of files will be output a lot, resulting OOM in Driver.

It can be solved by using ***distribute by*** on suitable cardinality(distinct values) column in the statement of Hive table creation.

distribute by clause limits number of hive table partitions. It considers cardinality of given column or **spark.sql.shuffle.partitions** which ever is minimal. For example, if **spark.sql.shuffle.partitions** is 200, but cardinality of column is 100, out files is 200, but the other 100 files are empty. So using very low cardinality column like 1 will cause data skew and will effect later query distribution.

So we suggest using the column with cardinality greater than **spark.sql.shuffle.partitions**. It can be greater than 2 to 3 times.

Example:

create table hivetable1 as select * from sourcetable1 distribute by col_age;

12.3.8.11 Why CarbonData tables created in V100R002C50RC1 not reflecting the privileges provided in Hive Privileges for non-owner?

Question

Why CarbonData tables created in V100R002C50RC1 not reflecting the privileges provided in Hive Privileges for non-owner?

Answer

The Hive ACL is implemented after the version V100R002C50RC1, hence the Hive ACL Privileges are not reflecting.

To support HIVE ACL Privileges for CarbonData tables created in V100R002C50RC1, following two ALTER TABLE commands must be executed by owner of the table.

ALTER TABLE *$dbname.$tablename SET LOCATION '$carbon.store/$dbname/$tablename';*

ALTER TABLE *$dbname.$tablename SET SERDEPROPERTIES ('path'='$carbon.store/$dbname/$tablename');*

Example:

Assume database name is 'carbondb', table name is 'carbontable', and CarbonData store location is 'hdfs://hacluster/user/hive/warehouse/carbon.store', then the commands should be executed is as follows:

ALTER TABLE *carbondb.carbontable SET LOCATION 'hdfs://hacluster/user/hive/warehouse/carbon.store/carbondb/carbontable';*

ALTER TABLE *carbondb.carbontable SET SERDEPROPERTIES ('path'='hdfs://hacluster/user/hive/warehouse/carbon.store/carbondb/carbontable');*

12.3.8.12 How Do I Logically Split Data Across Different Namespaces?

Question

How do I logically split data across different namespaces?

Answer

- Configuration:

  To logically split data across different namespaces, you must update the following configuration in the **core-site.xml** file of HDFS, Hive, and Spark.

  NOTE

  Changing the Hive component will change the locations of carbonstore and warehouse.

  – Configuration in HDFS

- **fs.defaultFS**: Name of the default file system. The URI mode must be set to **viewfs**. When **viewfs** is used, the permission part must be **ClusterX**.
- **fs.viewfs.mountable.ClusterX.homedir**: Home directory base path. You can use the getHomeDirectory() method defined in **FileSystem/FileContext** to access the home directory.
- fs.viewfs.mountable.default.link.<dir_name>: ViewFS mount table.

Example:

```
<property>
<name>fs.defaultFS</name>
<value>viewfs://ClusterX/</value>
</property>
<property>
<name>fs.viewfs.mounttable.ClusterX.link./folder1</name>
<value>hdfs://NS1/folder1</value>
</property>
<property>
<name>fs.viewfs.mounttable.ClusterX.link./folder2</name>
<value>hdfs://NS2/folder2</value>
</property>
```

– Configurations in Hive and Spark

fs.defaultFS: Name of the default file system. The URI mode must be set to **viewfs**. When **viewfs** is used, the permission part must be **ClusterX**.

- Syntax:

LOAD DATA INPATH *'path to data' INTO TABLE table_name OPTIONS ('...');*

NOTE

When Spark is configured with the viewFS file system and attempts to load data from HDFS, users must specify a path such as **viewfs://** or a relative path as the file path in the **LOAD** statement.

- Example:
  – Sample viewFS path:

  LOAD DATA INPATH *'viewfs://ClusterX/dir/data.csv' INTO TABLE table_name OPTIONS ('...');*

  – Sample relative path:

  LOAD DATA INPATH *'/apps/input_data1.txt'* ***INTO TABLE*** *table_name*;

12.3.8.13 Why Missing Privileges Exception is Reported When I Perform Drop Operation on Databases?

Question

Why drop database cascade is throwing the following exception?

```
Error: org.apache.spark.sql.AnalysisException: Missing Privileges;(State=,code=0)
```

Answer

This error is thrown when the owner of the database performs ***drop database <database_name> cascade*** which contains tables created by other users.

12.3.8.14 Why the UPDATE Command Cannot Be Executed in Spark Shell?

Question

Why the UPDATE command cannot be executed in Spark Shell?

Answer

The syntax and examples provided in this document are about Beeline commands instead of Spark Shell commands.

To run the UPDATE command in Spark Shell, use the following syntax:

- Syntax 1

  <carbon_context>.sql("UPDATE <CARBON TABLE> SET (column_name1, column_name2, ... column_name n) = (column1_expression , column2_expression , column3_expression ... column n_expression) [WHERE { <filter_condition> }];").show

- Syntax 2

  <carbon_context>.sql("UPDATE <CARBON TABLE> SET (column_name1, column_name2,) = (select sourceColumn1, sourceColumn2 from sourceTable [WHERE { <filter_condition> }]) [WHERE { <filter_condition> }];").show

Example:

If the context of CarbonData is **carbon**, run the following command:

carbon.sql("update carbonTable1 d set (d.column3,d.column5) = (select s.c33 ,s.c55 from sourceTable1 s where d.column1 = s.c11) where d.column1 = 'country' exists(select * from table3 o where o.c2 > 1);").show

12.3.8.15 How Do I Configure Unsafe Memory in CarbonData?

Question

How do I configure unsafe memory in CarbonData?

Answer

In the Spark configuration, the value of **spark.yarn.executor.memoryOverhead** must be greater than the sum of (**sort.inmemory.size.inmb** + **Netty offheapmemory required**), or the sum of (**carbon.unsafe.working.memory.in.mb** + **carbon.sort.inememory.storage.size.in.mb** + **Netty offheapmemory required**). Otherwise, if off-heap access exceeds the configured executor memory, Yarn may stop the executor.

If **spark.shuffle.io.preferDirectBufs** is set to **true**, the netty transfer service in Spark takes off some heap memory (around 384 MB or 0.1 x executor memory) from **spark.yarn.executor.memoryOverhead**.

For details, see **Configuring Executor Off-Heap Memory**.

12.3.8.16 Why Exception Occurs in CarbonData When Disk Space Quota is Set for Storage Directory in HDFS?

Question

Why exception occurs in CarbonData when Disk Space Quota is set for the storage directory in HDFS?

Answer

The data will be written to HDFS when you during create table, load table, update table, and so on. If the configured HDFS directory does not have sufficient disk space quota, then the operation will fail and throw following exception.

```
org.apache.hadoop.hdfs.protocol.DSQuotaExceededException:
The DiskSpace quota of /user/tenant is exceeded:
quota = 314572800 B = 300 MB but diskspace consumed = 402653184 B = 384 MB at
org.apache.hadoop.hdfs.server.namenode.DirectoryWithQuotaFeature.verifyStoragespaceQuota(DirectoryWithQuotaFeature.java:211) at
org.apache.hadoop.hdfs.server.namenode.DirectoryWithQuotaFeature.verifyQuota(DirectoryWithQuotaFeature.java:239) at
org.apache.hadoop.hdfs.server.namenode.FSDirectory.verifyQuota(FSDirectory.java:941) at
org.apache.hadoop.hdfs.server.namenode.FSDirectory.updateCount(FSDirectory.java:745)
```

If such exception occurs, configure a sufficient disk space quota for the tenant.

For example:

If the HDFS replication factor is 3 and HDFS default block size is 128 MB, then at least 384 MB (no. of block x block_size x replication_factor of the schema file = 1 x 128 x 3 = 384 MB) disk space quota is required to write a table schema file to HDFS.

NOTE

In case of fact files, as the default block size is 1024 MB, the minimum space required is 3072 MB per fact file for data load.

12.3.8.17 Why Does Data Query or Loading Fail and "org.apache.carbondata.core.memory.MemoryException: Not enough memory" Is Displayed?

Question

Why does data query or loading fail and "org.apache.carbondata.core.memory.MemoryException: Not enough memory" is displayed?

Answer

This exception is thrown when the out-of-heap memory required for data query and loading in the executor is insufficient.

In this case, increase the values of **carbon.unsafe.working.memory.in.mb** and **spark.yarn.executor.memoryOverhead**.

For details, see **How Do I Configure Unsafe Memory in CarbonData?**.

The memory is shared by data query and loading. Therefore, if the loading and query operations need to be performed at the same time, you are advised to set **carbon.unsafe.working.memory.in.mb** and **spark.yarn.executor.memoryOverhead** to a value greater than 2,048 MB.

The following formula can be used for estimation:

Memory required for data loading:

carbon.number.of.cores.while.loading [default value is 6] x Number of tables to load in parallel x offheap.sort.chunk.size.inmb [default value is 64 MB] + carbon.blockletgroup.size.in.mb [default value is 64 MB] + Current compaction ratio [64 MB/3.5])

= Around 900 MB per table

Memory required for data query:

(SPARK_EXECUTOR_INSTANCES. [default value is 2] x (carbon.blockletgroup.size.in.mb [default value: 64 MB] + carbon.blockletgroup.size.in.mb [default value = 64 MB x 3.5) x Number of cores per executor [default value: 1])

= ~ 600 MB

12.3.8.18 Why Do Files of a Carbon Table Exist in the Recycle Bin Even If the drop table Command Is Not Executed When Mis-deletion Prevention Is Enabled?

Question

Why do files of a Carbon table exist in the recycle bin even if the **drop table** command is not executed when mis-deletion prevention is enabled?

Answer

After the the mis-deletion prevention is enabled for a Carbon table, calling a file deletion command will move the deleted files to the recycle bin. The intermediate file **.carbonindex** is deleted durtion the execution of the **insert** or **load** command. Therefore, the table files may exist in the recycle bin even through the **drop table** command is not executed. If you run the **drop table** command, a table directory with a timestamp is generated. The files in the directory are complete.

12.4 Using ClickHouse

12.4.1 Using ClickHouse from Scratch

ClickHouse is a column-based database oriented to online analysis and processing. It supports SQL query and provides good query performance. The aggregation analysis and query performance based on large and wide tables is excellent, which is one order of magnitude faster than other analytical databases.

Prerequisites

You have installed the client, for example, in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory. Before using the client, download and update the client configuration file, and ensure that the active management node of Manager is available.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create ClickHouse tables. For details about how to bind a role to the user, see **ClickHouse User and Permission Management**. If Kerberos authentication is disabled for the current cluster, skip this step.

1. Run the following command if it is an MRS 3.1.0 cluster:

   export CLICKHOUSE_SECURITY_ENABLED=true

2. **kinit** *Component service user*

   Example: **kinit clickhouseuser**

Step 5 Run the client command of the ClickHouse component.

Run the **clickhouse -h** command to view the command help of ClickHouse.

The command output is as follows:

```
Use one of the following commands:
clickhouse local [args]
clickhouse client [args]
clickhouse benchmark [args]
clickhouse server [args]
clickhouse performance-test [args]
clickhouse extract-from-config [args]
clickhouse compressor [args]
clickhouse format [args]
clickhouse copier [args]
clickhouse obfuscator [args]
...
```

Run the **clickhouse client** command to connect to the ClickHouse serverif MRS 3.1.0 or later.

- Using SSL for login when Kerberos authentication is disabled for the current cluster:

  clickhouse client --host *IP address of the ClickHouse instance* **--user** *Username* **--password** *Password* **--port** 9440 **--secure**

- Using SSL for login when Kerberos authentication is enabled for the current cluster:

You must create a user on Manager because there is no default user. For details, see **ClickHouse User and Permission Management**.

After the user authentication is successful, you do not need to carry the **--user** and **--password** parameters when logging in to the client as the authenticated user.

clickhouse client --host *IP address of the ClickHouse instance* **--port** 9440 **--secure**

The following table describes the parameters of the **clickhouse client** command.

Table 12-54 Parameters of the **clickhouse client** command

Parameter	Description
--host	Host name of the server. The default value is **localhost**. You can use the host name or IP address of the node where the ClickHouse instance is located. **NOTE** You can log in to FusionInsight Manager and choose **Cluster** > **Services** > **ClickHouse** > **Instance** to obtain the service IP address of the ClickHouseServer instance.
--port	Port for connection. • If the SSL security connection is used, the default port number is **9440**, the parameter **--secure** must be carried. For details about the port number, search for the **tcp_port_secure** parameter in the ClickHouseServer instance configuration. • If non-SSL security connection is used, the default port number is **9000**, the parameter **--secure** does not need to be carried. For details about the port number, search for the **tcp_port** parameter in the ClickHouseServer instance configuration.
--user	Username. You can create the user on Manager and bind a role to the user. For details, see **ClickHouse User and Permission Management**. • If Kerberos authentication is enabled for the current cluster and the user authentication is successful, you do not need to carry the **--user** and **--password** parameters when logging in to the client as the authenticated user. You must create a user with this name on Manager because there is no default user in the Kerberos cluster scenario. • If Kerberos authentication is not enabled for the current cluster, you can specify a user and its password created on Manager when logging in to the client. If the user and password parameters are not carried, user **default** is used for login by default.
--password	Password. The default password is an empty string. This parameter is used together with the **--user** parameter. You can set a password when creating a user on Manager.
--query	Query to process when using non-interactive mode.

Parameter	Description
--database	Current default database. The default value is **default**, which is the default configuration on the server.
--multiline	If this parameter is specified, multiline queries are allowed. (**Enter** only indicates line feed and does not indicate that the query statement is complete.)
--multiquery	If this parameter is specified, multiple queries separated with semicolons (;) can be processed. This parameter is valid only in non-interactive mode.
--format	Specified default format used to output the result.
--vertical	If this parameter is specified, the result is output in vertical format by default. In this format, each value is printed on a separate line, which helps to display a wide table.
--time	If this parameter is specified, the query execution time is printed to **stderr** in non-interactive mode.
--stacktrace	If this parameter is specified, stack trace information will be printed when an exception occurs.
--config-file	Name of the configuration file.
--secure	If this parameter is specified, the server will be connected in SSL mode.
--history_file	Path of files that record command history.
--param_<name>	Query with parameters. Pass values from the client to the server. For details, see **https://clickhouse.tech/docs/en/interfaces/cli/#cli-queries-with-parameters**.

----End

12.4.2 ClickHouse Table Engine Overview

Background

Table engines play a key role in ClickHouse to determine:

- Where to write and read data
- Supported query modes
- Whether concurrent data access is supported
- Whether indexes can be used
- Whether multi-thread requests can be executed
- Parameters used for data replication

This section describes MergeTree and Distributed engines, which are the most important and frequently used ClickHouse table engines.

For details about other table engines, visit **https://clickhouse.tech/docs/en/engines/table-engines**.

MergeTree Family

Engines of the MergeTree family are the most universal and functional table engines for high-load tasks. They have the following key features:

- Data is stored by partition and block based on partitioning keys.
- Data index is sorted based on primary keys and the **ORDER BY** sorting keys.
- Data replication is supported by table engines prefixed with Replicated.
- Data sampling is supported.

When data is written, a table with this type of engine divides data into different folders based on the partitioning key. Each column of data in the folder is an independent file. A file that records serialized index sorting is created. This structure reduces the volume of data to be retrieved during data reading, greatly improving query efficiency.

- MergeTree

  Syntax for creating a table:

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1] [TTL expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2] [TTL expr2],
    ...
    INDEX index_name1 expr1 TYPE type1(...) GRANULARITY value1,
    INDEX index_name2 expr2 TYPE type2(...) GRANULARITY value2
) ENGINE = MergeTree()
ORDER BY expr
[PARTITION BY expr]
[PRIMARY KEY expr]
[SAMPLE BY expr]
[TTL expr [DELETE|TO DISK 'xxx'|TO VOLUME 'xxx'], ...]
[SETTINGS name=value, ...]
```

  Example:

```
CREATE TABLE default.test (
  name1 DateTime,
  name2 String,
  name3 String,
  name4 String,
  name5 Date,
  ...
) ENGINE = MergeTree()
PARTITION BY toYYYYMM(name5)
ORDER BY (name1, name2)
SETTINGS index_granularity = 8192
```

  Parameters in the example are described as follows:

  - **ENGINE = MergeTree()**: specifies the MergeTree engine.
  - **PARTITION BY toYYYYMM(name4)**: specifies the partition. The sample data is partitioned by month, and a folder is created for each month.
  - **ORDER BY**: specifies the sorting field. A multi-field index can be sorted. If the first field is the same, the second field is used for sorting, and so on.
  - **index_granularity = 8192**: specifies the index granularity. One index value is recorded for every 8,192 data records.

  If the data to be queried exists in a partition or sorting field, the data query time can be greatly reduced.

- ReplacingMergeTree

  Different from MergeTree, ReplacingMergeTree deletes duplicate entries with the same sorting key. ReplacingMergeTree is suitable for clearing duplicate data to save space, but it does not guarantee the absence of duplicate data. Generally, it is not recommended.

  Syntax for creating a table:

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = ReplacingMergeTree([ver])
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

- SummingMergeTree

  When merging data parts in SummingMergeTree tables, ClickHouse merges all rows with the same primary key into one row that contains summed values for the columns with the numeric data type. If the primary key is composed in a way that a single key value corresponds to large number of rows, storage volume can be significantly reduced and the data query speed can be accelerated.

  Syntax for creating a table:

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = SummingMergeTree([columns])
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

  Example:

  Create a SummingMergeTree table named **testTable**.

```
CREATE TABLE testTable
(
    id UInt32,
    value UInt32
)
ENGINE = SummingMergeTree()
ORDER BY id
```

  Insert data into the table.

```
INSERT INTO testTable Values(5,9),(5,3),(4,6),(1,2),(2,5),(1,4),(3,8);
INSERT INTO testTable Values(88,5),(5,5),(3,7),(3,5),(1,6),(2,6),(4,7),(4,6),(43,5),(5,9),(3,6);
```

  Query all data in unmerged parts.

```
SELECT * FROM testTable
┌─id─┬─value─┐
│  1 │     6 │
│  2 │     5 │
│  3 │     8 │
│  4 │     6 │
│  5 │    12 │
└────┴───────┘

┌─id─┬─value─┐
│  1 │     6 │
│  2 │     6 │
│  3 │    18 │
```

```
┌─id─┬─value─┐
│  4 │    13 │
│  5 │    14 │
│ 43 │     5 │
│ 88 │     5 │
└────┴───────┘
```

If ClickHouse has not summed up all rows and you need to aggregate data by ID, use the **sum** function and **GROUP BY** statement.

```
SELECT id, sum(value) FROM testTable GROUP BY id
┌─id─┬─sum(value)─┐
│  4 │         19 │
│  3 │         26 │
│ 88 │          5 │
│  2 │         11 │
│  5 │         26 │
│  1 │         12 │
│ 43 │          5 │
└────┴────────────┘
```

Merge rows manually.

```
OPTIMIZE TABLE testTable
```

Query data in the **testTable** table again.

```
SELECT * FROM testTable
┌─id─┬─value─┐
│  1 │    12 │
│  2 │    11 │
│  3 │    26 │
│  4 │    19 │
│  5 │    26 │
│ 43 │     5 │
│ 88 │     5 │
└────┴───────┘
```

SummingMergeTree uses the **ORDER BY** sorting keys as the condition keys to aggregate data. That is, if sorting keys are the same, data records are merged into one and the specified merged fields are aggregated.

Data is pre-aggregated only when merging is executed in the background, and the merging execution time cannot be predicted. Therefore, it is possible that some data has been pre-aggregated and some data has not been aggregated. Therefore, the **GROUP BY** statement must be used during aggregation.

- AggregatingMergeTree

  AggregatingMergeTree is a pre-aggregation engine used to improve aggregation performance. When merging partitions, the AggregatingMergeTree engine aggregates data based on predefined conditions, calculates data based on predefined aggregate functions, and saves the data in binary format to tables.

  Syntax for creating a table:

  ```
  CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
  (
      name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
      name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
      ...
  ) ENGINE = AggregatingMergeTree()
  [PARTITION BY expr]
  [ORDER BY expr]
  [SAMPLE BY expr]
  [TTL expr]
  [SETTINGS name=value, ...]
  ```

  Example:

You do not need to set the AggregatingMergeTree parameter separately. When partitions are merged, data in each partition is aggregated based on the **ORDER BY** sorting key. You can set the aggregate functions to be used and column fields to be calculated by defining the AggregateFunction type, as shown in the following example:

```
create table test_table (
name1 String,
name2 String,
name3 AggregateFunction(uniq,String),
name4 AggregateFunction(sum,Int),
name5 DateTime
) ENGINE = AggregatingMergeTree()
PARTITION BY toYYYYMM(name5)
ORDER BY (name1,name2)
PRIMARY KEY name1;
```

When data of the AggregateFunction type is written or queried, the ***state** and ***merge** functions need to be called. The asterisk (*) indicates the aggregate functions used for defining the field type. For example, the **uniq** and **sum** functions are specified for the **name3** and **name4** fields defined in the **test_table**, respectively. Therefore, you need to call the **uniqState** and **sumState** functions and run the **INSERT** and **SELECT** statements when writing data into the table.

```
insert into test_table select '8','test1',uniqState('name1'),sumState(toInt32(100)),'2021-04-30 17:18:00';
insert into test_table select '8','test1',uniqState('name1'),sumState(toInt32(200)),'2021-04-30 17:18:00';
```

When querying data, you need to call the corresponding functions **uniqMerge** and **sumMerge**.

```
select name1,name2,uniqMerge(name3),sumMerge(name4) from test_table group by name1,name2;
┌─name1─┬─name2─┬─uniqMerge(name3)─┬─sumMerge(name4)─┐
│ 8     │ test1 │                1 │             300 │
└───────┴───────┴──────────────────┴─────────────────┘
```

AggregatingMergeTree is more commonly used with materialized views, which are query views of other data tables at the upper layer. For details, visit **https://clickhouse.tech/docs/en/engines/table-engines/mergetree-family/aggregatingmergetree/**

- CollapsingMergeTree

  CollapsingMergeTree defines a **Sign** field to record status of data rows. If **Sign** is **1**, the data in this row is valid. If **Sign** is **-1**, the data in this row needs to be deleted.

  Syntax for creating a table:

  ```
  CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
  (
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
  ) ENGINE = CollapsingMergeTree(sign)
  [PARTITION BY expr]
  [ORDER BY expr]
  [SAMPLE BY expr]
  [SETTINGS name=value, ...]
  ```

  Example:

  For details about the example, visit **https://clickhouse.tech/docs/en/engines/table-engines/mergetree-family/collapsingmergetree/**.

- VersionedCollapsingMergeTree

The VersionedCollapsingMergeTree engine adds **Version** to the table creation statement to record the mapping between a **state** row and a **cancel** row in case that rows are out of order. The rows with the same primary key, same **Version**, and opposite **Sign** will be deleted during compaction.

Syntax for creating a table:

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = VersionedCollapsingMergeTree(sign, version)
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

Example:

For details about the example, visit **https://clickhouse.tech/docs/en/engines/table-engines/mergetree-family/versionedcollapsingmergetree/**.

- GraphiteMergeTree

  The GraphiteMergeTree engine is used to store data in the time series database Graphite.

  Syntax for creating a table:

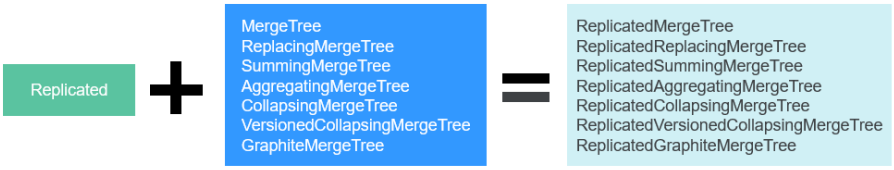
  ```
  CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
  (
      Path String,
      Time DateTime,
      Value <Numeric_type>,
      Version <Numeric_type>
      ...
  ) ENGINE = GraphiteMergeTree(config_section)
  [PARTITION BY expr]
  [ORDER BY expr]
  [SAMPLE BY expr]
  [SETTINGS name=value, ...]
  ```

  Example:

  For details about the example, visit **https://clickhouse.tech/docs/en/engines/table-engines/mergetree-family/graphitemergetree/**.

Replicated*MergeTree Engines

All engines of the MergeTree family in ClickHouse prefixed with Replicated become MergeTree engines that support replicas.



Replicated series engines use ZooKeeper to synchronize data. When a replicated table is created, all replicas of the same shard are synchronized based on the information registered with ZooKeeper.

Template for creating a Replicated engine:

```
ENGINE = Replicated*MergeTree('Storage path in ZooKeeper','Replica name', ...)
```

Two parameters need to be specified for a Replicated engine:

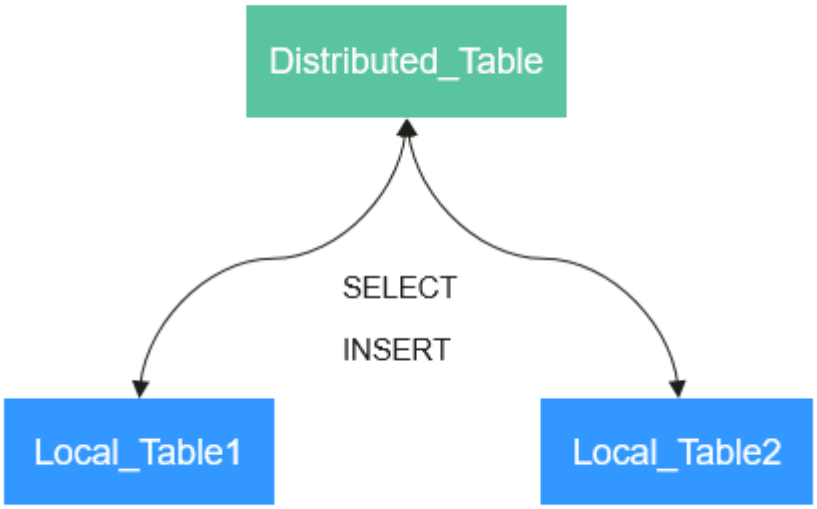
- *Storage path in ZooKeeper*: specifies the path for storing table data in ZooKeeper. The path format is ***/clickhouse/tables/{shard}|Database name/Table name***.
- *Replica name*: Generally, **{replica}** is used.

For details about the example, see **Creating a ClickHouse Table**.

Distributed Engine

The Distributed engine does not store any data. It serves as a transparent proxy for data shards and can automatically transmit data to each node in the cluster. Distributed tables need to work with other local data tables. Distributed tables distribute received read and write tasks to each local table where data is stored.

Figure 12-5 Working principle of the Distributed engine



Template for creating a Distributed engine:

```
ENGINE = Distributed(cluster_name, database_name, table_name, [sharding_key])
```

Parameters of a distributed table are described as follows:

- **cluster_name**: specifies the cluster name. When a distributed table is read or written, the cluster configuration information is used to search for the corresponding ClickHouse instance node.
- **database_name**: specifies the database name.
- **table_name**: specifies the name of a local table in the database. It is used to map a distributed table to a local table.
- **sharding_key** (optional): specifies the sharding key, based on which a distributed table distributes data to each local table.

Example:

```
-- Create a ReplicatedMergeTree local table named test.
CREATE TABLE default.test ON CLUSTER default_cluster_1
(
    `EventDate` DateTime,
    `id` UInt64
)
ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/default/test', '{replica}')
PARTITION BY toYYYYMM(EventDate)
ORDER BY id

-- Create a distributed table named test_all based on the local table test.
CREATE TABLE default.test_all ON CLUSTER default_cluster_1
(
    `EventDate` DateTime,
    `id` UInt64
)
ENGINE = Distributed(default_cluster_1, default, test, rand())
```

Rules for creating a distributed table:

- When creating a distributed table, add **ON CLUSTER** *cluster_name* to the table creation statement so that the statement can be executed once on a ClickHouse instance and then distributed to all instances in the cluster for execution.
- Generally, a distributed table is named in the following format: *Local table name*_all. It forms a one-to-many mapping with local tables. Then, multiple local tables can be operated using the distributed table proxy.
- Ensure that the structure of a distributed table is the same as that of local tables. If they are inconsistent, no error is reported during table creation, but an exception may be reported during data query or insertion.

12.4.3 Creating a ClickHouse Table

ClickHouse implements the replicated table mechanism based on the ReplicatedMergeTree engine and ZooKeeper. When creating a table, you can specify an engine to determine whether the table is highly available. Shards and replicas of each table are independent of each other.

ClickHouse also implements the distributed table mechanism based on the Distributed engine. Views are created on all shards (local tables) for distributed query, which is easy to use. ClickHouse has the concept of data sharding, which is one of the features of distributed storage. That is, parallel read and write are used to improve efficiency.

The ClickHouse cluster table engine that uses Kunpeng as the CPU architecture does not support HDFS and Kafka.

Viewing cluster and Other Environment Parameters of ClickHouse

Step 1 Use the ClickHouse client to connect to the ClickHouse server by referring to **Using ClickHouse from Scratch**.

Step 2 Query the cluster identifier and other information about the environment parameters.

select cluster,shard_num,replica_num,host_name from system.clusters;

```
SELECT
    cluster,
```

```
    shard_num,
    replica_num,
    host_name
FROM system.clusters

┌─cluster─────────────┬─shard_num─┬─replica_num─┬─host_name──────────┐
│ default_cluster_1   │         1 │           1 │ node-master1dOnG   │
│ default_cluster_1   │         1 │           2 │ node-group-1tXED0001 │
│ default_cluster_1   │         2 │           1 │ node-master2OXQS   │
│ default_cluster_1   │         2 │           2 │ node-group-1tXED0002 │
│ default_cluster_1   │         3 │           1 │ node-master3QsRI   │
│ default_cluster_1   │         3 │           2 │ node-group-1tXED0003 │
└─────────────────────┴───────────┴─────────────┴────────────────────┘

6 rows in set. Elapsed: 0.001 sec.
```

Step 3 Query the shard and replica identifiers.

select * from system.macros;

```
SELECT *
FROM system.macros

┌─macro───┬─substitution──────┐
│ id      │ 76                │
│ replica │ node-master3QsRI  │
│ shard   │ 3                 │
└─────────┴───────────────────┘

3 rows in set. Elapsed: 0.001 sec.
```

----End

Creating a Local Replicated Table and a distributed Table

Step 1 Log in to the ClickHouse node using the client, for example, **clickhouse client --host** *node-master3QsRI* **--multiline --port 9440 --secure;**

NOTE

node-master3QsRI is the value of **host_name** obtained in **Step 2** in **Viewing cluster and Other Environment Parameters of ClickHouse**.

Step 2 Create a replicated table using the ReplicatedMergeTree engine.

For details about the syntax, see **https://clickhouse.tech/docs/en/engines/table-engines/mergetree-family/replication/#creating-replicated-tables**.

For example, run the following commands to create a ReplicatedMergeTree table named **test** on the **default_cluster_1** node and in the **default** database:

CREATE TABLE *default.test* **ON CLUSTER** *default_cluster_1*

(

`EventDate` DateTime,

`id` UInt64

)

ENGINE = ReplicatedMergeTree('*/clickhouse/tables/{shard}/default/test***', '***{replica}***')**

PARTITION BY toYYYYMM(EventDate)

ORDER BY id;

The parameters are described as follows:

- The **ON CLUSTER** syntax indicates the distributed DDL, that is, the same local table can be created on all instances in the cluster after the statement is executed once.
- **default_cluster_1** is the cluster identifier obtained in **Step 2** in **Viewing cluster and Other Environment Parameters of ClickHouse**.

> **CAUTION**
>
> **ReplicatedMergeTree** engine receives the following two parameters:
>
> – Storage path of the table data in ZooKeeper
>
> The path must be in the **/clickhouse** directory. Otherwise, data insertion may fail due to insufficient ZooKeeper quota.
>
> To avoid data conflict between different tables in ZooKeeper, the directory must be in the following format:
>
> */clickhouse/tables/{shard}/default/test*, in which **/clickhouse/tables/{shard}** is fixed, *default* indicates the database name, and *text* indicates the name of the created table.
>
> – Replica name: Generally, **{replica}** is used.

```
CREATE TABLE default.test ON CLUSTER default_cluster_1
(
   `EventDate` DateTime,
   `id` UInt64
)
ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/default/test', '{replica}')
PARTITION BY toYYYYMM(EventDate)
ORDER BY id

┌─host──────────────────────┬─port─┬─status─┬─error─┬─num_hosts_remaining─┬─num_hosts_activ
e─┐
│ node-group-1tXED0002      │ 9000 │      0 │       │                   5 │           3 │
│ node-group-1tXED0003      │ 9000 │      0 │       │                   4 │           3 │
│ node-master1dOnG          │ 9000 │      0 │       │                   3 │           3 │
└───────────────────────────┴──────┴────────┴───────┴─────────────────────┴───────────────┘
┌─host──────────────────────┬─port─┬─status─┬─error─┬─num_hosts_remaining─┬─num_hosts_activ
e─┐
│ node-master3QsRI          │ 9000 │      0 │       │                   2 │           0 │
│ node-group-1tXED0001      │ 9000 │      0 │       │                   1 │           0 │
│ node-master2OXQS          │ 9000 │      0 │       │                   0 │           0 │
└───────────────────────────┴──────┴────────┴───────┴─────────────────────┴───────────────┘

6 rows in set. Elapsed: 0.189 sec.
```

Step 3 Create a distributed table using the Distributed engine.

For example, run the following commands to create a distributed table named **test_all** on the **default_cluster_1** node and in the **default** database:

CREATE TABLE *default.test_all* **ON CLUSTER** *default_cluster_1*

(

`EventDate` DateTime,

`id` UInt64

)

ENGINE = Distributed(*default_cluster_1, default, test, rand()***);**

```
CREATE TABLE default.test_all ON CLUSTER default_cluster_1
(
   `EventDate` DateTime,
   `id` UInt64
)
ENGINE = Distributed(default_cluster_1, default, test, rand())

┌─host──────────────────────┬─port─┬─status─┬─error─┬─num_hosts_remaining─┬─num_hosts_active─┐
│ node-group-1tXED0002      │ 9000 │      0 │       │                   5 │                0 │
│ node-master3QsRI          │ 9000 │      0 │       │                   4 │                0 │
│ node-group-1tXED0003      │ 9000 │      0 │       │                   3 │                0 │
│ node-group-1tXED0001      │ 9000 │      0 │       │                   2 │                0 │
│ node-master1dOnG          │ 9000 │      0 │       │                   1 │                0 │
│ node-master2OXQS          │ 9000 │      0 │       │                   0 │                0 │
└───────────────────────────┴──────┴────────┴───────┴─────────────────────┴──────────────────┘

6 rows in set. Elapsed: 0.115 sec.
```

NOTE

Distributed requires the following parameters:

- **default_cluster_1** is the cluster identifier obtained in **Step 2** in **Viewing cluster and Other Environment Parameters of ClickHouse**.
- **default** indicates the name of the database where the local table is located.
- **test** indicates the name of the local table. In this example, it is the name of the table created in **Step 2**.
- (Optional) Sharding key

  This key and the weight configured in the **config.xml** file determine the route for writing data to the distributed table, that is, the physical table to which the data is written. It can be the original data (for example, **site_id**) of a column in the table or the result of the function call, for example, **rand()** is used in the preceding SQL statement. Note that data must be evenly distributed in this key. Another common operation is to use the hash value of a column with a large difference, for example, **intHash64(user_id)**.

----End

ClickHouse Table Data Operations

Step 1 Log in to the ClickHouse node on the client. Example:

clickhouse client --host *node-master3QsRI* **--multiline --port 9440 --secure;**

NOTE

node-master3QsRI is the value of **host_name** obtained in **Step 2** in **Viewing cluster and Other Environment Parameters of ClickHouse**.

Step 2 After creating a table by referring to **Creating a Local Replicated Table and a distributed Table**, you can insert data to the local table.

For example, run the following command to insert data to the local table **test**:

insert into test values(toDateTime(now()), rand());

Step 3 Query the local table information.

For example, run the following command to query data information of the table **test** in **Step 2**:

select * from test;

```
SELECT *
FROM test

┌─────────────EventDate─┬──────────id─┐
│ 2020-11-05 21:10:42   │ 1596238076  │
└───────────────────────┴─────────────┘

1 rows in set. Elapsed: 0.002 sec.
```

Step 4 Query the distributed table.

For example, the distributed table **test_all** is created based on table **test** in **Step 3**. Therefore, the same data in table **test** can also be queried in table **test_all**.

select * from test_all;

```
SELECT *
FROM test_all

┌─────────────EventDate─┬──────────id─┐
│ 2020-11-05 21:10:42   │ 1596238076  │
└───────────────────────┴─────────────┘

1 rows in set. Elapsed: 0.004 sec.
```

Step 5 Switch to the shard node with the same **shard_num** and query the information about the current table. The same table data can be queried.

For example, run the **exit;** command to exit the original node.

Run the following command to switch to the **node-group-1tXED0003** node:

clickhouse client --host *node-group-1tXED0003* **--multiline --port 9440 --secure;**

NOTE

The **shard_num** values of **node-group-1tXED0003** and **node-master3QsRI** are the same by performing **Step 2**.

show tables;

```
SHOW TABLES

┌─name─────┐
│ test     │
│ test_all │
└──────────┘
```

Step 6 Query the local table data. For example, run the following command to query data in table **test** on the **node-group-1tXED0003** node:

select * from test;

```
SELECT *
FROM test

┌─────────────EventDate─┬──────────id─┐
│ 2020-11-05 21:10:42   │ 1596238076  │
└───────────────────────┴─────────────┘
```

```
1 rows in set. Elapsed: 0.005 sec.
```

Step 7 Switch to the shard node with different **shard_num** value and query the data of the created table.

For example, run the following command to exit the **node-group-1tXED0003** node:

exit;

Switch to the **node-group-1tXED0001** node. The **shard_num** values of **node-group-1tXED0001** and **node-master3QsRI** are different by performing **Step 2**.

clickhouse client --host *node-group-1tXED0001* **--multiline --port 9440 --secure;**

Query the local table **test**. Data cannot be queried on the different shard node because table **test** is a local table.

select * from test;

```
SELECT *
FROM test

Ok.
```

Query data in the distributed table **test_all**. The data can be queried properly.

select * from test_all;

```
SELECT *
FROM test

┌─────────────EventDate─┬──────────id─┐
│ 2020-11-05 21:12:19   │ 3686805070  │
└───────────────────────┴─────────────┘

1 rows in set. Elapsed: 0.002 sec.
```

----End

12.4.4 Common ClickHouse SQL Syntax

12.4.4.1 CREATE DATABASE: Creating a Database

This section describes the basic syntax and usage of the SQL statement for creating a ClickHouse database.

Basic Syntax

CREATE DATABASE [IF NOT EXISTS] *Database_name* **[ON CLUSTER** *ClickHouse cluster name***]**

NOTE

The syntax **ON CLUSTER** *ClickHouse cluster name* enables the Data Definition Language (DDL) statement to be executed on all instances in the cluster at a time. You can run the following statement to obtain the cluster name from the **cluster** field:

select cluster,shard_num,replica_num,host_name from system.clusters;

Example

```
-- Create a database named test.
CREATE DATABASE test ON CLUSTER default_cluster;
-- After the creation is successful, run the query command for verification.
show databases;
┌─name────┐
│ default │
│ system  │
│ test    │
└─────────┘
```

12.4.4.2 CREATE TABLE: Creating a Table

This section describes the basic syntax and usage of the SQL statement for creating a ClickHouse table.

Basic Syntax

- Method 1: Creating a table named **table_name** in the specified **database_name** database.

  If the table creation statement does not contain **database_name**, the name of the database selected during client login is used by default.

  CREATE TABLE [IF NOT EXISTS] *[database_name.]table_name* **[ON CLUSTER** *ClickHouse cluster name***]**

  (

  name1 **[***type1***] [DEFAULT|MATERIALIZED|ALIAS** *expr1***],**

  name2 **[***type2***] [DEFAULT|MATERIALIZED|ALIAS** *expr2***],**

  ...

) *ENGINE = engine_name()*

  [**PARTITION BY** *expr_list*]

  [**ORDER BY** *expr_list*]

  CAUTION

  You are advised to use **PARTITION BY** to create table partitions when creating a ClickHouse table. The ClickHouse data migration tool migrates data based on table partitions. If you do not use **PARTITION BY** to create table partitions during table creation, the table data cannot be migrated on the GUI in **Using the ClickHouse Data Migration Tool**.

- Method 2: Creating a table with the same structure as **database_name2.table_name2** and specifying a different table engine for the table

If no table engine is specified, the created table uses the same table engine as **database_name2.table_name2**.

CREATE TABLE [IF NOT EXISTS] *[database_name.]table_name* **AS** [*database_name2*.]*table_name*2 [ENGINE = *engine_name*]

- Method 3: Using the specified engine to create a table with the same structure as the result of the **SELECT** clause and filling it with the result of the **SELECT** clause

  CREATE TABLE [IF NOT EXISTS] *[database_name.]table_name ENGINE = engine_name* **AS SELECT** ...

Example

```
-- Create a table named test in the default database and default_cluster cluster.
CREATE TABLE default.test ON CLUSTER default_cluster
(
    `EventDate` DateTime,
    `id` UInt64
)
ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/default/test', '{replica}')
PARTITION BY toYYYYMM(EventDate)
ORDER BY id
```

12.4.4.3 INSERT INTO: Inserting Data into a Table

This section describes the basic syntax and usage of the SQL statement for inserting data to a table in ClickHouse.

Basic Syntax

- Method 1: Inserting data in standard format

  INSERT INTO *[database_name.]table* [(*c1, c2, c3*)] **VALUES** (*v11, v12, v13*), (*v21, v22, v23*), ...

- Method 2: Using the **SELECT** result to insert data

  INSERT INTO *[database_name.]table* [(c1, c2, c3)] **SELECT** ...

Example

```
-- Insert data into the test2 table.
insert into test2 (id, name) values (1, 'abc'), (2, 'bbbb');
-- Query data in the test2 table.
select * from test2;
┌─id─┬─name─┐
│  1 │ abc  │
│  2 │ bbbb │
└────┴──────┘
```

12.4.4.4 SELECT: Querying Table Data

This section describes the basic syntax and usage of the SQL statement for querying table data in ClickHouse.

Basic Syntax

SELECT [**DISTINCT**] expr_list

[**FROM** [*database_name*.]*table* | (subquery) | table_function] [**FINAL**]

[SAMPLE sample_coeff]

[ARRAY **JOIN** ...]

[**GLOBAL**] [**ANY**|**ALL**|**ASOF**] [**INNER**|**LEFT**|**RIGHT**|**FULL**|**CROSS**] [**OUTER**|SEMI|ANTI] **JOIN** (subquery)|**table** (**ON** <expr_list>)|(**USING** <column_list>)

[PREWHERE expr]

[**WHERE** expr]

[**GROUP BY** expr_list] [**WITH** TOTALS]

[**HAVING** expr]

[**ORDER BY** expr_list] [**WITH** FILL] [**FROM** expr] [**TO** expr] [STEP expr]

[**LIMIT** [offset_value,]n **BY** columns]

[**LIMIT** [n,]m] [**WITH** TIES]

[**UNION ALL** ...]

[**INTO** OUTFILE filename]

[FORMAT format]

Example

```
-- View ClickHouse cluster information.
select * from system.clusters;
-- View the macros set for the current node.
select * from system.macros;
-- Check the database capacity.
select
sum(rows) as "Total number of rows",
formatReadableSize(sum(data_uncompressed_bytes)) as "Original size",
formatReadableSize(sum(data_compressed_bytes)) as "Compression size",
round(sum(data_compressed_bytes) / sum(data_uncompressed_bytes) * 100,
0) "Compression rate"
from system.parts;
-- Query the capacity of the test table. Add or modify the where clause based on the site requirements.
select
sum(rows) as "Total number of rows",
formatReadableSize(sum(data_uncompressed_bytes)) as "Original size",
formatReadableSize(sum(data_compressed_bytes)) as "Compression size",
round(sum(data_compressed_bytes) / sum(data_uncompressed_bytes) * 100,
0) "Compression rate"
from system.parts
where table in ('test')
and partition like '2020-11-%'
group by table;
```

12.4.4.5 ALTER TABLE: Modifying a Table Structure

This section describes the basic syntax and usage of the SQL statement for modifying a table structure in ClickHouse.

Basic Syntax

ALTER TABLE [*database_name*].*name* [**ON CLUSTER** *cluster*] **ADD**|**DROP**|**CLEAR**|**COMMENT**|**MODIFY COLUMN** ...

NOTE

ALTER supports only *MergeTree, Merge, and Distributed engine tables.

Example

```
-- Add the test01 column to the t1 table.
ALTER TABLE t1 ADD COLUMN test01 String DEFAULT 'defaultvalue';
-- Query the modified table t1.
desc t1
┌─name────┬─type───┬─default_type─┬─default_expression─┬─comment─┬─codec_expression─┬─ttl_expression─┐
│ id      │ UInt8  │              │                    │         │                  │                │
│ name    │ String │              │                    │         │                  │                │
│ address │ String │              │                    │         │                  │                │
│ test01  │ String │ DEFAULT      │ 'defaultvalue'     │         │                  │                │
└─────────┴────────┴──────────────┴────────────────────┴─────────┴──────────────────┴────────────────┘
-- Change the type of the name column in the t1 table to UInt8.
ALTER TABLE t1 MODIFY COLUMN name UInt8;
-- Query the modified table t1.
desc t1
┌─name────┬─type───┬─default_type─┬─default_expression─┬─comment─┬─codec_expression─┬─ttl_expression─┐
│ id      │ UInt8  │              │                    │         │                  │                │
│ name    │ UInt8  │              │                    │         │                  │                │
│ address │ String │              │                    │         │                  │                │
│ test01  │ String │ DEFAULT      │ 'defaultvalue'     │         │                  │                │
└─────────┴────────┴──────────────┴────────────────────┴─────────┴──────────────────┴────────────────┘
-- Delete the test01 column from the t1 table.
ALTER TABLE t1 DROP COLUMN test01;
-- Query the modified table t1.
desc t1
┌─name────┬─type───┬─default_type─┬─default_expression─┬─comment─┬─codec_expression─┬─ttl_expression─┐
│ id      │ UInt8  │              │                    │         │                  │                │
│ name    │ UInt8  │              │                    │         │                  │                │
│ address │ String │              │                    │         │                  │                │
└─────────┴────────┴──────────────┴────────────────────┴─────────┴──────────────────┴────────────────┘
```

12.4.4.6 DESC: Querying a Table Structure

This section describes the basic syntax and usage of the SQL statement for querying a table structure in ClickHouse.

Basic Syntax

DESC|**DESCRIBE TABLE** [*database_name*.]*table* [**INTO** OUTFILE filename] [FORMAT format]

Example

```
-- Query the t1 table structure.
desc t1;
┌─name────┬─type───┬─default_type─┬─default_expression─┬─comment─┬─codec_expression─┬─ttl_expression─┐
│ id      │ UInt8  │              │                    │         │                  │                │
│ name    │ UInt8  │              │                    │         │                  │                │
│ address │ String │              │                    │         │                  │                │
└─────────┴────────┴──────────────┴────────────────────┴─────────┴──────────────────┴────────────────┘
```

12.4.4.7 DROP: Deleting a Table

This section describes the basic syntax and usage of the SQL statement for deleting a ClickHouse table.

Basic Syntax

DROP [**TEMPORARY**] **TABLE** [**IF EXISTS**] [*database_name*.]*name* [**ON CLUSTER** *cluster*]

Example

```
-- Delete the t1 table.
drop table t1;
```

12.4.4.8 SHOW: Displaying Information About Databases and Tables

This section describes the basic syntax and usage of the SQL statement for displaying information about databases and tables in ClickHouse.

Basic Syntax

show databases

show tables

Example

```
-- Query database information.
show databases;
┌─name───┐
│ default │
│ system │
│ test │
└────────┘
-- Query table information.
show tables;
┌─name──┐
│ t1 │
│ test │
│ test2 │
│ test5 │
└───────┘
```

12.4.5 Migrating ClickHouse Data

12.4.5.1 Using ClickHouse to Import and Export Data

Using ClickHouse to Import and Export Data

This section describes the basic syntax and usage of the SQL statements for importing and exporting file data using ClickHouse.

- Importing data in CSV format

  clickhouse client --host *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **--secure --format_csv_delimiter="***CSV file delimiter***" --query="INSERT INTO** *Table name* **FORMAT CSV" <** *Host path where the CSV file is stored*

  Example

  ```
  clickhouse client --host 10.5.208.5 --database testdb --port 21427 --secure --format_csv_delimiter="," --query="INSERT INTO testdb.csv_table FORMAT CSV" < /opt/data.csv
  ```

  You need to create a table in advance.

- Exporting data in CSV format

  CAUTION

  Exporting data files in CSV format may cause CSV injection. Exercise caution when performing this operation.

  clickhouse client --host *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **-m --secure --query=**"SELECT * **FROM** *Table name*" > *CSV file export path*

  Example

  ```
  clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="SELECT * FROM test_table" > /opt/test.csv
  ```

- Importing data in Parquet format

  cat *Parquet file* **| clickhouse client --host** *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **-m --secure --query="INSERT INTO** *Table name* **FORMAT Parquet"**

  Example

  ```
  cat /opt/student.parquet | clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="INSERT INTO parquet_tab001 FORMAT Parquet"
  ```

- Exporting data in Parquet format

  clickhouse client --host *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **-m --secure --query="select * from** *Table name* **FORMAT Parquet**" > *Parquet file export path*

  Example

  ```
  clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="select * from test_table FORMAT Parquet" > /opt/student.parquet
  ```

- Importing data in ORC format

  cat *ORC file path* **| clickhouse client --host** *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **-m --secure --query="INSERT INTO** *Table name* **FORMAT ORC"**

  Example

  ```
  cat /opt/student.orc | clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="INSERT INTO orc_tab001 FORMAT ORC"
  # Data in the ORC file can be exported from HDFS. For example:
  hdfs dfs -cat /user/hive/warehouse/hivedb.db/emp_orc/000000_0_copy_1 | clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="INSERT INTO orc_tab001 FORMAT ORC"
  ```

- Exporting data in ORC format

  clickhouse client --host *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **-m --secure --query="select * from** *Table name* **FORMAT ORC**" > *ORC file export path*

  Example

  ```
  clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="select * from csv_tab001 FORMAT ORC" > /opt/student.orc
  ```

- Importing data in JSON format

  INSERT INTO *Table name* **FORMAT JSONEachRow** *JSON string 1 JSON string 2*

  Example

```
INSERT INTO test_table001 FORMAT JSONEachRow {"PageViews":5,
"UserID":"4324182021466249494", "Duration":146,"Sign":-1}
{"UserID":"4324182021466249494","PageViews":6,"Duration":185,"Sign":1}
```

- Exporting data in JSON format

  clickhouse client --host *Host name or IP address of the ClickHouse instance* **--database** *Database name* **--port** *Port number* **-m --secure --query="SELECT** * **FROM** *Table name* **FORMAT JSON|JSONEachRow|JSONCompact|...**" > *JSON file export path*

  Example

```
# Export JSON file.
clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="SELECT *
FROM test_table FORMAT JSON" > /opt/test.json

# Export json(JSONEachRow).
clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="SELECT *
FROM test_table FORMAT JSONEachRow" > /opt/test_jsoneachrow.json

# Export json(JSONCompact).
clickhouse client --host 10.5.208.5 --database testdb --port 21427 -m --secure --query="SELECT *
FROM test_table FORMAT JSONCompact" > /opt/test_jsoncompact.json
```

12.4.5.2 Synchronizing Kafka Data to ClickHouse

This section describes how to create a Kafka table to automatically synchronize Kafka data to the ClickHouse cluster.

Prerequisites

- You have created a Kafka cluster. The Kafka client has been installed.
- A ClickHouse cluster has been created. It is in the same VPC as the Kafka cluster and can communicate with each other.
- The ClickHouse client has been installed.

Syntax of the Kafka Table

- **Syntax**

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER cluster]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = Kafka()
SETTINGS
    kafka_broker_list = 'host1:port1,host2:port2',
    kafka_topic_list = 'topic1,topic2,...',
    kafka_group_name = 'group_name',
    kafka_format = 'data_format';
    [kafka_row_delimiter = 'delimiter_symbol',]
    [kafka_schema = '',]
    [kafka_num_consumers = N]
```

- **Parameter description**

Table 12-55 Kafka table parameters

Parameter	Mandatory	Description
kafka_broker_list	Yes	A list of Kafka broker instances, separated by comma (,). For example, *IP address 1 of the Kafka broker instance*:**9092**,*IP address 2 of the Kafka broker instance*:**9092**,*IP address 3 of the Kafka broker instance*:**9092**. To obtain the IP address of the Kafka broker instance, perform the following steps: • For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **Kafka**. Click **Instances** to query the IP addresses of the Kafka instances. **NOTE** If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.) • For MRS 3.*x* or later, log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka**. Click **Instances** to query the IP addresses of the Kafka instances.
kafka_topic_list	Yes	A list of Kafka topics.
kafka_group_name	Yes	A group of Kafka consumers, which can be customized.
kafka_format	Yes	Kafka message format, for example, JSONEachRow, CSV, and XML.
kafka_row_delimiter	No	Delimiter character, which ends a message.
kafka_schema	No	Parameter that must be used if the format requires a schema definition.
kafka_num_consumers	No	Number of consumers in per table. The default value is **1**. If the throughput of a consumer is insufficient, more consumers are required. The total number of consumers cannot exceed the number of partitions in a topic because only one consumer can be allocated to each partition.

How to Synchronize Kafka Data to ClickHouse

Step 1 Switch to the Kafka client installation directory. For details, see **Using the Kafka Client**.

1. Log in to the node where the Kafka client is installed as the Kafka client installation user.
2. Run the following command to go to the client installation directory:

   cd /opt/client
3. Run the following command to configure environment variables:

   source bigdata_env
4. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step.
   a. Run the following command first for an MRS 3.1.0 cluster:

      export CLICKHOUSE_SECURITY_ENABLED=true
   b. **kinit** *Component service user*

Step 2 Run the following command to create a Kafka topic. For details, see **Managing Kafka Topics**.

kafka-topics.sh --topic *kafkacktest2* **--create --zookeeper** *IP address of the Zookeeper role instance:2181*/**kafka --partitions** *2* **--replication-factor** *1*

NOTE

- **--topic** is the name of the topic to be created, for example, **kafkacktest2**.
- **--zookeeper** is the IP address of the node where the ZooKeeper role instances are located, which can be the IP address of any of the three role instances. You can obtain the IP address of the node by performing the following steps:
  - For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **ZooKeeper** > **Instances**. View the IP addresses of the ZooKeeper role instances.
  - For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Instance**. View the IP addresses of the ZooKeeper role instances.
- **--partitions** and **--replication-factor** are the topic partitions and topic backup replicas, respectively. The number of the two parameters cannot exceed the number of Kafka role instances.

Step 3 Log in to the ClickHouse client by referring to **Using ClickHouse from Scratch**.

1. Run the following command to go to the client installation directory:

   cd /opt/Bigdata/client
2. Run the following command to configure environment variables:

   source bigdata_env
3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The user must have the permission to create ClickHouse tables. Therefore, you need to bind the corresponding role to the user. For details, see **ClickHouse User and Permission Management**. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *Component service user*

   Example: **kinit clickhouseuser**
4. Run the following command to connect to the ClickHouse instance node to which data is to be imported:

clickhouse client --host *IP address of the ClickHouse instance* **--user** *Login username* **--password** *User password* **--port** *Port number of the ClickHouse instance* **--database** *Database name* **--multiline**

Step 4 Create a Kafka table in ClickHouse by referring to **Syntax of the Kafka Table**. For example, the following table creation statement is used to create a Kafka table whose name is **kafka_src_tbl3**, topic name is **kafkacktest2**, and message format is **JSONEachRow** in the default database.

```
create table kafka_src_tbl3 on cluster default_cluster
(id UInt32, age UInt32, msg String)
ENGINE=Kafka()
SETTINGS
 kafka_broker_list='IP address 1 of the Kafka broker instance:9092,IP address 2 of the Kafka broker instance:9092,IP address 3 of the Kafka broker instance:9092',
 kafka_topic_list='kafkacktest2',
 kafka_group_name='cg12',
 kafka_format='JSONEachRow';
```

Step 5 Create a ClickHouse replicated table, for example, the ReplicatedMergeTree table named **kafka_dest_tbl3**.

```
create table kafka_dest_tbl3 on cluster default_cluster
( id UInt32, age UInt32, msg String )
engine = ReplicatedMergeTree('/clickhouse/tables/{shard}/default/kafka_dest_tbl3', '{replica}')
partition by age
order by id;
```

Step 6 Create a materialized view, which converts data in Kafka in the background and saves the data to the created ClickHouse table.

```
create materialized view consumer3 on cluster default_cluster to kafka_dest_tbl3 as select * from
kafka_src_tbl3;
```

Step 7 Perform **Step 1** again to go to the Kafka client installation directory.

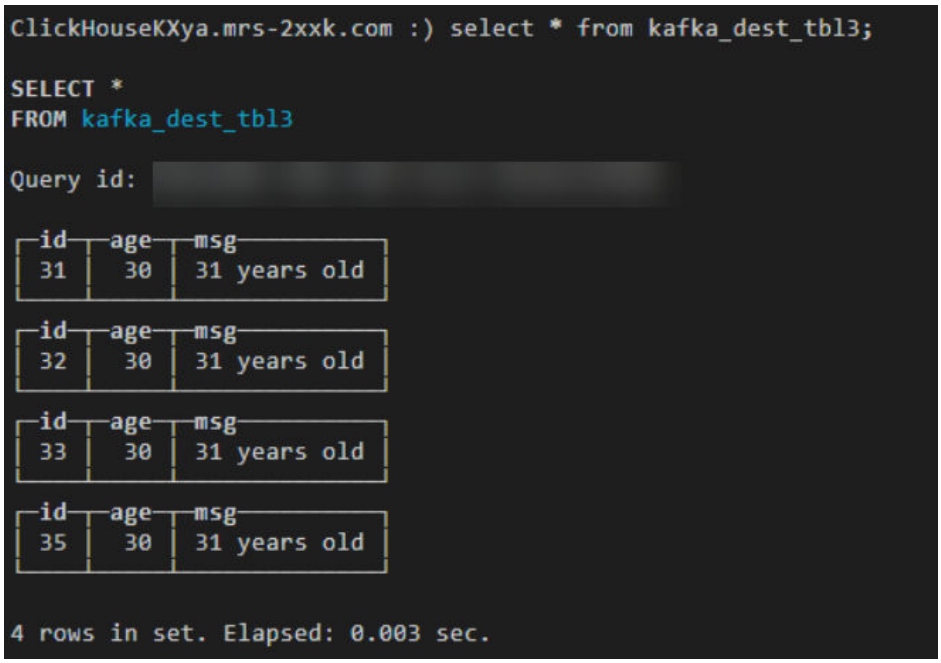
Step 8 Run the following command to send a message to the topic created in **Step 2**:

kafka-console-producer.sh --broker-list *IP address 1 of the kafka broker instance*:**9092,***IP address 2 of the kafka broker instance*:**9092,***IP address 3 of the kafka broker instance*:**9092 --topic** *kafkacktest2*

```
>{"id":31, "age":30, "msg":"31 years old"}
>{"id":32, "age":30, "msg":"31 years old"}
>{"id":33, "age":30, "msg":"31 years old"}
>{"id":35, "age":30, "msg":"31 years old"}
```

Step 9 Use the ClickHouse client to log in to the ClickHouse instance node in **Step 3** and query the ClickHouse table data, for example, to query the replicated table **kafka_dest_tbl3**. It shows that the data in the Kafka message has been synchronized to this table.

```
select * from kafka_dest_tbl3;
```



----End

12.4.5.3 Using the ClickHouse Data Migration Tool

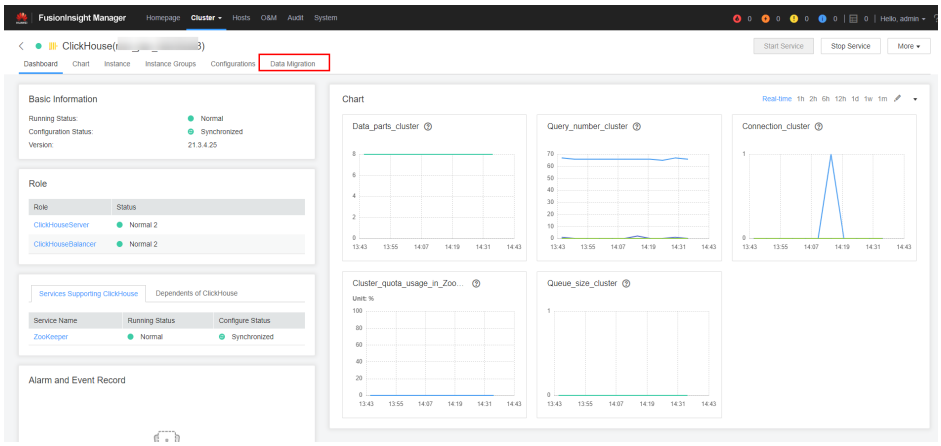
The ClickHouse data migration tool can migrate some partitions of one or more partitioned MergeTree tables on several ClickHouseServer nodes to the same tables on other ClickHouseServer nodes. In the capacity expansion scenario, you can use this tool to migrate data from an original node to a new node to balance data after capacity expansion.

Prerequisites

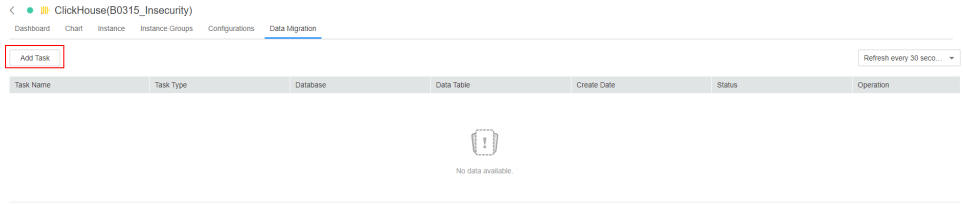
- The ClickHouse and Zookeeper services are running properly. The ClickHouseServer instances on the source and destination nodes are normal.
- The destination node has the data table to be migrated and the table is a partitioned MergeTree table.
- Before creating a migration task, ensure that all tasks for writing data to a table to be migrated have been stopped. After the task is started, you can only query the table to be migrated and cannot write data to or delete data from the table. Otherwise, data may be inconsistent before and after the migration.
- The ClickHouse data directory on the destination node has sufficient space.

Procedure

Step 1 Log in to Manager and choose **Cluster** > **Services** > **ClickHouse**. On the ClickHouse service page, click the **Data Migration** tab.



Step 2 Click **Add Task**.



Step 3 On the page for creating a migration task, set the migration task parameters. For details, see **Table 12-56**.

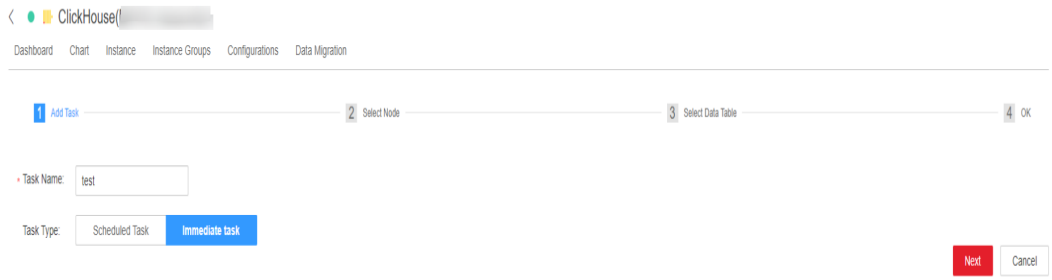
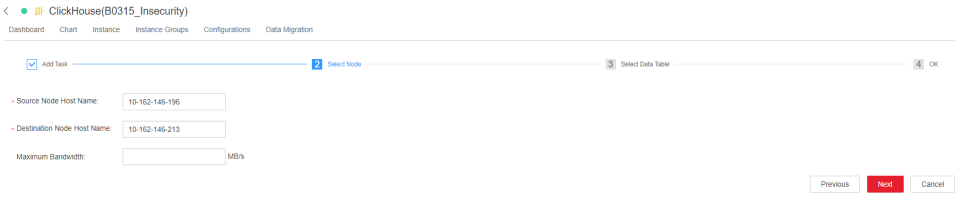


Table 12-56 Migration task parameters

Parameter	Description
Task Name	Enter a specific task name. The value can contain 1 to 50 characters, including letters, arrays, and underscores (_), and cannot be the same as that of an existing migration task.
Task Type	• **Scheduled Task**: When the scheduled task is selected, you can set **Started** to specify a time point later than the current time to execute the task. • **Immediate task**: The task is executed immediately after it is started.
Started	Set this parameter when **Task Type** is set to **Scheduled Task**. The valid value is a time point within 90 days from now.

Step 4 On the **Select Node** page, specify **Source Node Host Name** and **Destination Node Host Name**, and click **Next**.

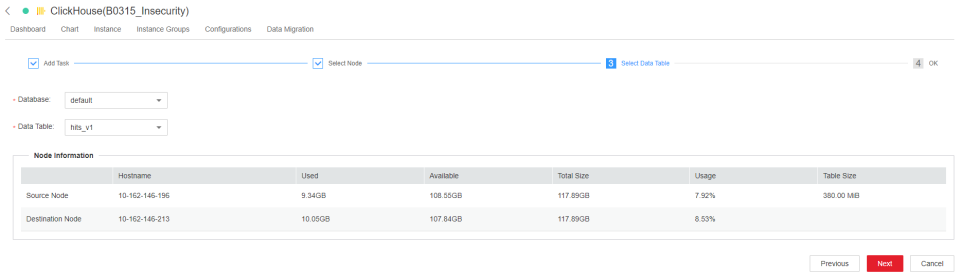


NOTE

- Only one host name can be entered in **Source Node Host Name** and **Destination Node Host Name**, respectively. Multi-node migration is not supported.

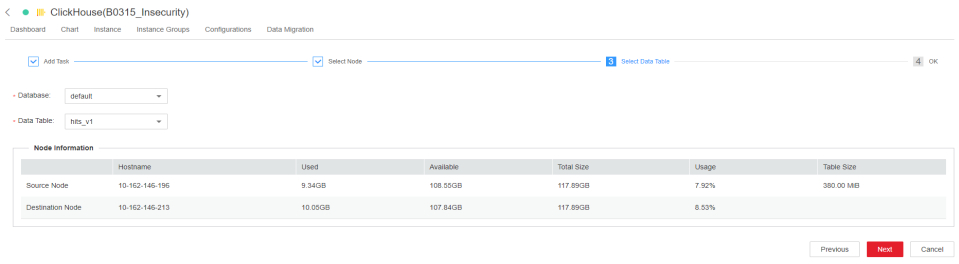
  To obtain the parameter values, click the **Instance** tab on the ClickHouse service page and view the **Host Name** column of the current ClickHouseServer instance.
- **Maximum Bandwidth** is optional. If it is not specified, there is no upper limit. The maximum bandwidth can be set to **10000** MB/s.

Step 5 On the **Select Data Table** page, click next to **Database**, select the database to be migrated on the source node, and select the data table to be migrated for **Data Table**. The data table drop-down list displays the partitioned MergeTree tables in the selected database. In the **Node Information** area, the space usage of the ClickHouse service data directory on the current source and destination nodes is displayed. Click **Next**.



Step 6 Confirm the task information and click **Submit**.

The data migration tool automatically calculates the partitions to be migrated based on the size of the data table. The amount of data to be migrated is the total size of the partitions to be migrated.



Step 7 After the migration task is submitted, click **Start** in the **Operation** column. If the task is an immediate task, the task starts to be executed. If the task is a scheduled task, the countdown starts.

Step 8 During the migration task execution, you can click **Cancel** to cancel the migration task that is being executed. If you cancel the task, the migrated data on the destination node will be rolled back.

You can choose **More** > **Details** to view the log information during the migration.

Step 9 After the migration is complete, choose **More** > **Results** to view the migration result and choose **More** > **Delete** to delete the directories related to the migration task on ZooKeeper and the source node.

----End

12.4.6 User Management and Authentication

12.4.6.1 ClickHouse User and Permission Management

User Permission Model

ClickHouse user permission management enables unified management of users, roles, and permissions on each ClickHouse instance in the cluster. You can use the permission management module of the Manager UI to create users, create roles, and bind the ClickHouse access permissions. User permissions are controlled by binding roles to users.

Resource management: **Table 12-57** lists the resources supported by ClickHouse permission management.

Resource permissions: **Table 12-58** lists the resource permissions supported by ClickHouse.

Table 12-57 Permission management objects supported by ClickHouse

Resource	Integration	Remarks
Database	Yes (level 1)	-
Table	Yes (level 2)	-
View	Yes (level 2)	Same as tables

Table 12-58 Resource permission list

Resource	Available Permission	Remarks
Database	CREATE	CREATE DATABASE/TABLE/VIEW/ DICTIONARY

Resource	Available Permission	Remarks
Table/View	SELECT/INSERT	-

Prerequisites

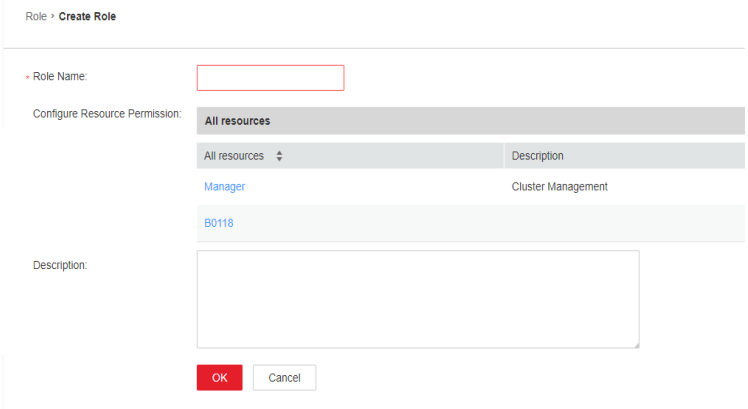
- The ClickHouse and Zookeeper services are running properly.
- When creating a database or table in the cluster, the **ON CLUSTER** statement is used to ensure that the metadata of the database and table on each ClickHouse node is the same.

NOTE

After the permission is granted, it takes about 1 minute for the permission to take effect.

Adding the ClickHouse Role

Step 1 Log in to Manager and choose **System** > **Permission** > **Role**. On the **Role** page, click **Create Role**.



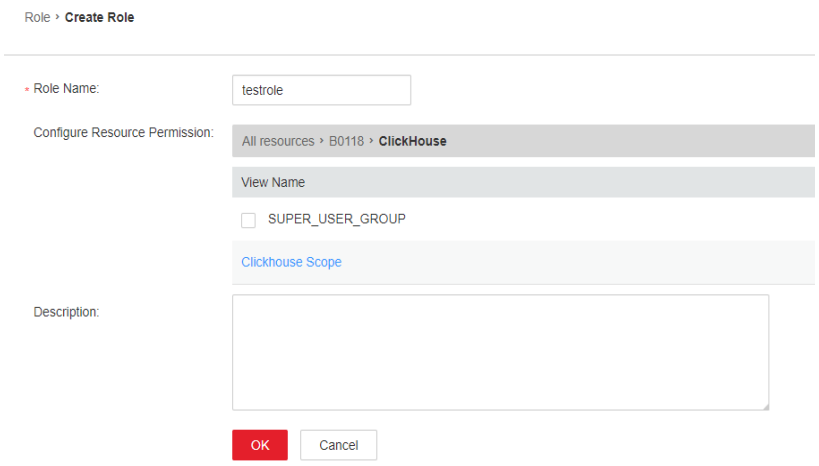
Step 2 On the **Create Role** page, specify **Role Name**. In the **Configure Resource Permission** area, click the cluster name. On the service list page that is displayed, click the ClickHouse service.

Determine whether to create a role with administrator permission based on service requirements.

NOTE

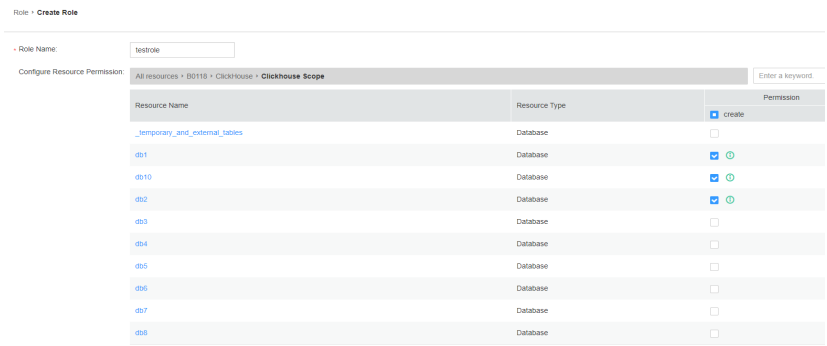
- The ClickHouse administrator has all the database operation permissions except the permissions to create, delete, and modify users and roles.
- Only the built-in user **clickhouse** of ClickHouse has the permission to manage users and roles.

- If yes, go to **Step 3**.
- If no, go to **Step 4**.



Step 3 Select **SUPER_USER_GROUP** and click **OK**.

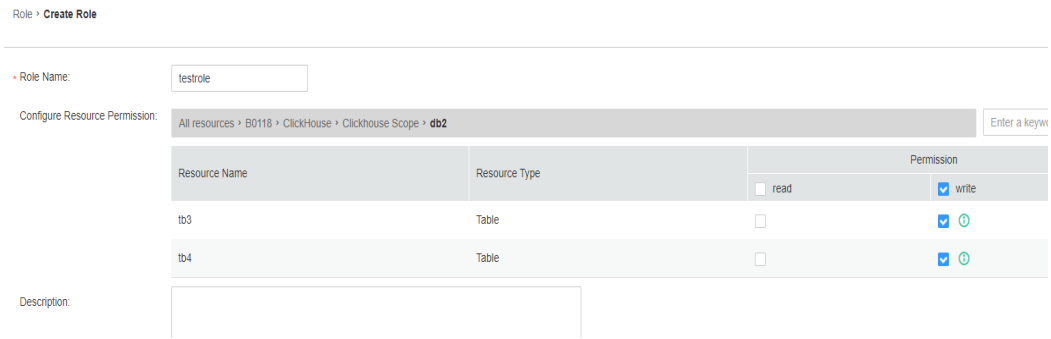
Step 4 Click **ClickHouse Scope**. The ClickHouse database resource list is displayed. If you select **create**, the role has the create permission on the database.



Determine whether to grant the permission based on the service requirements.

- If yes, click **OK**.
- If no, go to **Step 5**.

Step 5 Click the resource name and select the *Database resource name to be operated*. On the displayed page, select **READ** (SELECT permission) or **WRITE** (INSERT permission) based on service requirements, and click **OK**.



----End

Adding a User and Binding the ClickHouse Role to the User

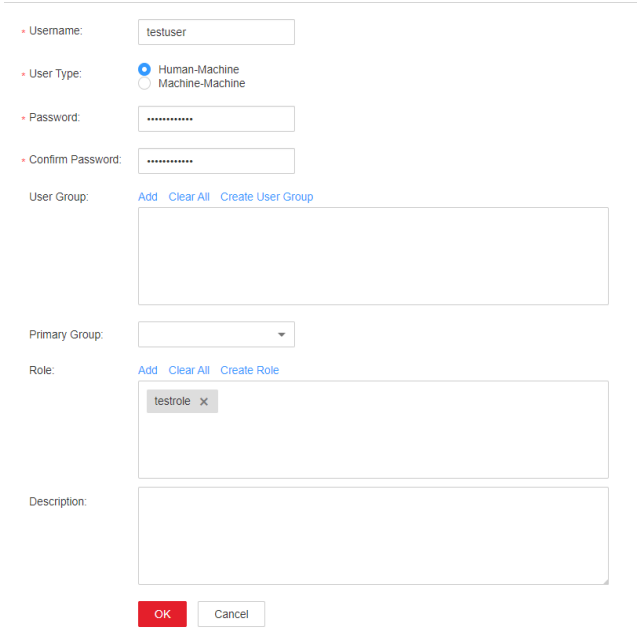
Step 1 Log in to Manager and choose **System** > **Permission** > **User** and click **Create**.

Step 2 Select **Human-Machine** for **User Type** and set **Password** and **Confirm Password** to the password of the user.

NOTE

- Username: The username cannot contain hyphens (-). Otherwise, the authentication will fail.
- Password: The password cannot contain special characters $, ., and #. Otherwise, the authentication will fail.

Step 3 In the **Role** area, click **Add**. In the displayed dialog box, select a role with the ClickHouse permission and click **OK** to add the role. Then, click **OK**.



Step 4 Log in to the node where the ClickHouse client is installed and use the new username and password to connect to the ClickHouse service.

- Run the following command to go to the client installation directory:

  cd /opt/*Client installation directory*

- Run the following command to configure environment variables:

  source bigdata_env

- If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The user must have the permission to create ClickHouse tables. Therefore, you need to bind the corresponding role to the user. For details, see **Adding the ClickHouse Role**. If Kerberos authentication is disabled for the current cluster, skip this step.

  a. Run the following command if it is an MRS 3.1.0 cluster:

  export CLICKHOUSE_SECURITY_ENABLED=true

  b. **kinit** *Component service user*

- Log in to the system as the new user.

  clickhouse client --host *ClickHouse instance IP* **--multiple --user** *User added in* ***Step 1*** **--password** *User password set in* ***Step 2*** **--port** *ClickHouse port number* **--secure**

----End

Granting Permissions Using the Client in Abnormal Scenarios

By default, the table metadata on each node of the ClickHouse cluster is the same. Therefore, the table information on a random ClickHouse node is collected on the permission management page of Manager. If the **ON CLUSTER** statement is not used when databases or tables are created on some nodes, the resource may fail to be displayed during permission management, and permissions may not be granted to the resource. To grant permissions on the local table on a single ClickHouse node, perform the following steps on the background client.

NOTE

The following operations are performed based on the obtained roles, database or table names, and IP addresses of the node where the corresponding ClickHouseServer instance is located.

- You can log in to FusionInsight Manager and choose **Cluster** > **Services** > **ClickHouse** > **Instance** to obtain the service IP address of the ClickHouseServer instance.
- The default system domain name is **hadoop.com**. Log in to FusionInsight Manager and choose **System** > **Permission** > **Domain and Mutual Trust**. The value of **Local Domain** is the system domain name. Change the letters to lowercase letters when running a command.

Step 1 Log in to the node where the ClickHouseServer instance is located as user **root**.

Step 2 Run the following command to obtain the path of the **clickhouse.keytab** file:

ls ${BIGDATA_HOME}/FusionInsight_ClickHouse_*/install/FusionInsight-ClickHouse-*/clickhouse/keytab/clickhouse.keytab

Step 3 Log in to the node where the client is installed as the client installation user.

Step 4 Run the following command to go to the client installation directory:

cd /opt/client

Step 5 Run the following command to configure environment variables:

source bigdata_env

Run the following command if it is an MRS 3.1.0 cluster with Kerberos authentication enabled:

export CLICKHOUSE_SECURITY_ENABLED=true

Step 6 Run the following command to connect to the ClickHouseServer instance:

If Kerberos authentication is enabled for the current cluster, run the following command:

clickhouse client --host *IP address of the node where the ClickHouseServer instance is located* **--user clickhouse/hadoop.**<*System domain name*> **--password** *clickhouse.keytab path obtained in* ***Step 2*** **--port** *ClickHouse port number* **--secure**

If Kerberos authentication is disabled for the current cluster, run the following command:

clickhouse client --host *IP address of the node where the ClickHouseServer instance is located* **--user clickhouse --port** *ClickHouse port number*

Step 7 Run the following statement to grant permissions to a database:

In the syntax for granting permissions, *DATABASE* indicates the name of the target database, and *role* indicates the target role.

GRANT [ON CLUSTER *cluster_name***]** *privilege* **ON** *{DATABASE|TABLE}* **TO** *{user | role]*

For example, grant user **testuser** the CREATE permission on database **t2**:

GRANT CREATE ON *m2* **to** *testuser***;**

Step 8 Run the following commands to grant permissions on the table or view. In the following command, *TABLE* indicates the name of the table or view to be operated, and *user* indicates the role to be operated.

Run the following command to grant the query permission on tables in a database:

GRANT SELECT ON *TABLE* **TO** *user;*

Run the following command to grant the write permission on tables in a database:

GRANT INSERT ON *TABLE* **TO** *user***;**

NOTE

For details about ClickHouse **GRANT** operations and permission description, visit **https://clickhouse.tech/docs/en/sql-reference/statements/grant/**.

Step 9 Run the following command to exit the client:

quit;

----End

12.4.6.2 Interconnecting ClickHouse With OpenLDAP for Authentication

ClickHouse can be interconnected with OpenLDAP. You can manage accounts and permissions in a centralized manner by adding the OpenLDAP server configuration and creating users on ClickHouse. You can use this method to import users from the OpenLDAP server to ClickHouse in batches.

This section applies only to MRS 3.1.0 or later.

Prerequisites

- The MRS cluster and ClickHouse instances are running properly, and the ClickHouse client has been installed.
- OpenLDAP has been installed and is running properly.

Creating a ClickHouse User for Interconnecting with the OpenLDAP Server

Step 1 Log in to Manager and choose **Cluster** > **Services** > **ClickHouse**. Click the **Configurations** tab and then **All Configurations**.

Step 2 Choose **ClickHouseServer(Role)** > **Customization**, and add the following OpenLDAP configuration parameters to the **clickhouse-config-customize** configuration item.

Table 12-59 OpenLDAP parameters

Parameter	Description	Example Value
ldap_servers.ldap_server_name.host	OpenLDAP server host name or IP address. This parameter cannot be empty.	localhost
ldap_servers.ldap_server_name.port	OpenLDAP server port number. If **enable_tls** is set to **true**, the default port number is **636**. Otherwise, the default port number is **389**.	636
ldap_servers.ldap_server_name.auth_dn_prefix	Prefix and suffix used to construct the DN to bind to. The generated DN will be constructed as a string in the following format: **auth_dn_prefix** + **escape(user_name)** + **auth_dn_suffix**. Use a comma (,) as the first non-space character of **auth_dn_suffix**.	uid=
ldap_servers.ldap_server_name.auth_dn_suffix		,ou=Group,dc=node1,dc=com
ldap_servers.ldap_server_name.enable_tls	A tag to trigger the use of the secure connection to the OpenLDAP server. • Set it to **no** for the plaintext (ldap://) protocol (not recommended). • Set it to **yes** for the LDAP over SSL/TLS (ldaps://) protocol.	yes
ldap_servers.ldap_server_name.tls_require_cert	SSL/TLS peer certificate verification behavior. The value can be **never**, **allow**, **try**, or **require**.	allow

NOTE

For details about other parameters, see **<ldap_servers> Parameters**.

Step 3 After the configuration is complete, click **Save**. In the displayed dialog box, click **OK**. After the configuration is saved, click **Finish**.

Step 4 On Manager, click **Instance**, select a ClickHouseServer instance, and choose **More** > **Restart Instance**. In the displayed dialog box, enter the password and click **OK**. In the displayed **Restart instance** dialog box, click **OK**. Confirm that the instance is restarted successfully as prompted and click **Finish**.

Step 5 Log in to the ClickHouseServer instance node and go to the **${BIGDATA_HOME}/FusionInsight_ClickHouse_***Version number*/*x_x*_**ClickHouseServer/etc** directory.

cd ${BIGDATA_HOME}/FusionInsight_ClickHouse_*/*x_x*_ClickHouseServer/etc

Step 6 Run the following command to view the **config.xml** configuration file and check whether the OpenLDAP parameters are configured successfully:

cat config.xml

```
[root@k                3 etc]# cat config.xml
<yandex>
  <ldap_servers>
    <ldap_server_name>
      <auth_dn_prefix>uid=</auth_dn_prefix>
      <port>636</port>
      <host>localhost</host>
      <enable_tls>yes</enable_tls>
      <tls_require_cert>allow</tls_require_cert>
      <auth_dn_suffix>,ou=Group,dc=node1,dc=com</auth_dn_suffix>
    </ldap_server_name>
  </ldap_servers>
  <zookeeper>
```

Step 7 Log in to the node where the ClickHouseServer instance is located as user **root**.

Step 8 Run the following command to obtain the path of the **clickhouse.keytab** file:

ls ${BIGDATA_HOME}/FusionInsight_ClickHouse_*/install/FusionInsight-ClickHouse-*/clickhouse/keytab/clickhouse.keytab

Step 9 Log in to the node where the client is installed as the client installation user.

Step 10 Run the following command to go to the ClickHouse client installation directory:

cd /opt/client

Step 11 Run the following command to configure environment variables:

source bigdata_env

Step 12 Run the following command to connect to the ClickHouseServer instance:

- If Kerberos authentication is enabled for the current cluster, use **clickhouse.keytab** to connect to the ClickHouseServer instance.

  clickhouse client --host *IP address of the node where the ClickHouseServer instance is located* **--user clickhouse/hadoop.***<System domain name>* **--password** *clickhouse.keytab path obtained in* ***Step 8*** **--port** *ClickHouse port number*

NOTE

The default system domain name is **hadoop.com**. Log in to FusionInsight Manager and choose **System** > **Permission** > **Domain and Mutual Trust**. The value of **Local Domain** is the system domain name. Change the letters to lowercase letters when running a command.

- If Kerberos authentication is disabled for the current cluster, connect to the ClickHouseServer instance as the **clickhouse** administrator.

  clickhouse client --host *IP address of the node where the ClickHouseServer instance is located* **--user clickhouse --port** *ClickHouse port number*

Step 13 Create a common user of OpenLDAP.

Run the following statement to create user **testUser** in cluster **default_cluster** and set **ldap_server** to the OpenLDAP server name in the **<ldap_servers>** tag in **Step 6**. In this example, the name is **ldap_server_name**.

CREATE USER *testUser* **ON CLUSTER** *default_cluster* **IDENTIFIED WITH ldap_server BY** 'ldap_server_name'**;**

testUser indicates an existing username in OpenLDAP. Change it based on the site requirements.

Step 14 Log out of the client, and then log in to the client as the new user to check whether the configuration is successful.

exit;

clickhouse client --host *ClickHouseServer IP address* **--user** *testUser* **--password** *testUser password* **--port** *ClickHouse port number*

----End

<ldap_servers> Parameters

- **host**

  OpenLDAP server host name or IP address. This parameter is mandatory and cannot be empty.

- **port**

  Port number of the OpenLDAP server. If **enable_tls** is set to **true**, the default value is **636**. Otherwise, the value is **389**.

- **auth_dn_prefix, auth_dn_suffix**

  Prefix and suffix used to construct the DN to bind to.

  The generated DN will be constructed as a string in the following format: **auth_dn_prefix** + **escape(user_name)** + **auth_dn_suffix**.

  Note that you should use a comma (,) as the first non-space character of **auth_dn_suffix**.

- **enable_tls**

  A tag to trigger the use of the secure connection to the OpenLDAP server.

  Set it to **no** for the plaintext (ldap://) protocol (not recommended).

  Set it to **yes** for LDAP over SSL/TLS (ldaps://) protocol (recommended and default).

- **tls_minimum_protocol_version**
  Minimum protocol version of SSL/TLS.
  The value can be **ssl2**, **ssl3**, **tls1.0**, **tls1.1**, or **tls1.2** (default).
- **tls_require_cert**
  SSL/TLS peer certificate verification behavior.
  The value can be **never**, **allow**, **try**, or **require** (default).
- **tls_cert_file**
  Certificate file.
- **tls_key_file**
  Certificate key file.
- **tls_ca_cert_file**
  CA certificate file.
- **tls_ca_cert_dir**
  Directory where the CA certificate is stored.
- **tls_cipher_suite**
  Allowed encryption suite.

12.4.7 Backing Up and Restoring ClickHouse Data Using a Data File

Scenario

This section describes how to back up data by exporting ClickHouse data to a CSV file and restore data using the CSV file.

Prerequisites

- You have installed the ClickHouse client.
- You have created a user with related permissions on ClickHouse tables on Manager.
- You have prepared a server for backup.

Backing Up Data

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create ClickHouse tables. If Kerberos authentication is disabled for the current cluster, skip this step.

1. Run the following command if it is an MRS 3.1.0 cluster:

   export CLICKHOUSE_SECURITY_ENABLED=true

2. **kinit** *Component service user*

   Example: **kinit clickhouseuser**

Step 5 Run the ClickHouse client command to export the ClickHouse table data to be backed up to a specified directory.

clickhouse client --host *Host name or instance IP address* **--secure --port 21427 --query=**"*Table query statement*" > *Path of the exported CSV file*

The following shows an example of backing up data in the **test** table to the **default_test.csv** file on the ClickHouse instance **10.244.225.167**.

clickhouse client --host 10.244.225.167 --secure --port 21427 --query="select * from default.test FORMAT CSV" > /opt/clickhouse/default_test.csv

Step 6 Upload the exported CSV file to the backup server.

----End

Restoring Data

Step 1 Upload the backup data file on the backup server to the directory where the ClickHouse client is located.

For example, upload the **default_test.csv** backup file to the **/opt/clickhouse** directory.

Step 2 Log in to the node where the client is installed as the client installation user.

Step 3 Run the following command to go to the client installation directory:

cd /opt/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create ClickHouse tables. If Kerberos authentication is disabled for the current cluster, skip this step.

1. Run the following command if it is an MRS 3.1.0 cluster:

   export CLICKHOUSE_SECURITY_ENABLED=true

2. **kinit** *Component service user*

   Example: **kinit clickhouseuser**

Step 6 Run the ClickHouse client command to log in to the ClickHouse cluster.

clickhouse client --host *Host name or instance IP address* **--secure --port 21427**

Step 7 Create a table with the format corresponding to the CSV file.

CREATE TABLE [IF NOT EXISTS] *[database_name.]table_name* **[ON CLUSTER** *Cluster name***]**

(

name1 **[***type1***] [DEFAULT|***materialized***|ALIAS** *expr1***],**

name2 **[***type2***] [DEFAULT|***materialized***|ALIAS** *expr2***],**

...

) *ENGINE = engine*

Step 8 Import the content in the backup file to the table created in **Step 7** to restore data.

clickhouse client --host *Host name or instance IP address* **--secure --port 21427 --query="insert into** *Table name* **FORMAT CSV" <** *CSV file path*

The following shows an example of restoring data from the **default_test.csv** backup file to the **test_cpy** table on the ClickHouse instance **10.244.225.167**.

clickhouse client --host 10.244.225.167 --secure --port 21427 --query="insert into default.test_cpy FORMAT CSV" < /opt/clickhouse/default_test.csv

----End

12.4.8 ClickHouse Log Overview

Log Description

Log path: The default storage path of ClickHouse log files is as follows: **${BIGDATA_LOG_HOME}/clickhouse**

Log archive rule: The automatic ClickHouse log compression function is enabled. By default, when the size of logs exceeds 100 MB, logs are automatically compressed into a log file named in the following format: *<Original log name>***.***[ID]***.gz**. A maximum of 10 latest compressed files are reserved by default. The number of compressed files can be configured on Manager.

Table 12-60 ClickHouse log list

Log Type	Log File Name	Description
Run logs	/var/log/Bigdata/clickhouse/clickhouseServer/clickhouse-server.err.log	Path of ClickHouseServer error log files.
	/var/log/Bigdata/clickhouse/clickhouseServer/checkService.log	Path of key ClickHouseServer run log files.
	/var/log/Bigdata/clickhouse/clickhouseServer/clickhouse-server.log	
	/var/log/Bigdata/clickhouse/balance/start.log	Path of ClickHouseBalancer startup log files.
	/var/log/Bigdata/clickhouse/balance/error.log	Path of ClickHouseBalancer error log files.
	/var/log/Bigdata/clickhouse/balance/access_http.log	Path of ClickHouseBalancer run log files.

Log Type	Log File Name	Description
Data migration logs	/var/log/Bigdata/clickhouse/migration/*Data migration task name*/clickhouse-copier_{timestamp}_{processId}/copier.log	Run logs generated when you use the migration tool by referring to **Using the ClickHouse Data Migration Tool**.
	/var/log/Bigdata/clickhouse/migration/*Data migration task name*/clickhouse-copier_{timestamp}_{processId}/copier.err.log	Error logs generated when you use the migration tool by referring to **Using the ClickHouse Data Migration Tool**.

Log Level

Table 12-61 describes the log levels supported by ClickHouse.

Levels of run logs are error, warning, trace, information, and debug from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-61 Log levels

Log Type	Level	Description
Run log	error	Logs of this level record error information about system running.
	warning	Logs of this level record exception information about the current event processing.
	trace	Logs of this level record trace information about the current event processing.
	information	Logs of this level record normal running status information about the system and events.
	debug	Logs of this level record system running and debugging information.

To modify log levels, perform the following operations:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > **Services** > **ClickHouse** > **Configurations**.

Step 3 Select **All Configurations**.

Step 4 On the menu bar on the left, select the log menu of the target role.

Step 5 Select a desired log level.

Step 6 Click **Save**. Then, click **OK**.

----End

NOTE

The configurations take effect immediately without the need to restart the service.

Log Format

The following table lists the ClickHouse log format:

Table 12-62 Log formats

Log Type	Format	Example
Run log	*<yyyy-MM-dd HH:mm:ss,SSS>*|*<Log level>*|*<Name of the thread that generates the log>*|*<Message in the log>*|*<Location where the log event occurs>*	2021.02.23 15:26:30.691301 [6085] {} <Error> DynamicQueryHandler: Code: 516, e.displayText() = DB::Exception: default: Authentication failed: password is incorrect or there is no user with such name, Stack trace (when copying this message, always include the lines below): 0. Poco::Exception::Exception(std::__1::basic_string<char, std::__1::char_traits<char>, std::__1::allocator<char> > const&, int) @ 0x1250e59c

12.5 Using DBService

12.5.1 DBService Log Overview

Log Description

Log path: The default storage path of DBService log files is **/var/log/Bigdata/dbservice**.

- GaussDB: **/var/log/Bigdata/dbservice/DB** (GaussDB run log directory), **/var/log/Bigdata/dbservice/scriptlog/gaussdbinstall.log** (GaussDB installation log), and **/var/log/gaussdbuninstall.log** (GaussDB uninstallation log).
- HA: **/var/log/Bigdata/dbservice/ha/runlog** (HA run log directory) and **/var/log/Bigdata/dbservice/ha/scriptlog** (HA script log directory)
- DBServer: **/var/log/Bigdata/dbservice/healthCheck** (Directory of service and process health check logs)

  /var/log/Bigdata/dbservice/scriptlog (run log directory), **/var/log/Bigdata/audit/dbservice/** (audit log directory)

Log archive rule: The automatic DBService log compression function is enabled. By default, when the size of logs exceeds 1 MB, logs are automatically compressed into a log file named in the following format: *<Original log file name>-[No.]*.**gz**. A maximum of 20 latest compressed files are reserved.

NOTE

Log archive rules cannot be modified.

Table 12-63 DBService log list

Type	Log File Name	Description
DBServer run log	dbservice_serviceCheck.log	Run log file of the service check script
	dbservice_processCheck.log	Run log file of the process check script
	backup.log	Run logs of backup and restoration operations (The DBService backup and restoration operations need to be performed.)
	checkHaStatus.log	Log file of HA check records
	cleanupDBService.log	Uninstallation log file (You need to uninstall DBService logs.)

Type	Log File Name	Description
	componentUserManager.log	Log file that records the adding and deleting operations on the database by users (Services that depend on DBService need to be added.)
	install.log	Installation log file
	preStartDBService.log	Pre-startup log file
	start_dbserver.log	DBServer startup operation log file (DBService needs to be started.)
	stop_dbserver.log	DBServer stop operation log file (DBService needs to be stopped.)
	status_dbserver.log	Log file of the DBServer status check (You need to execute the **$DBSERVICE_HOME/sbin/status-dbserver.sh** script.)
	modifyPassword.log	Run log file of changing the DBService password script. (You need to execute the **$DBSERVICE_HOME/sbin/modifyDBPwd.sh** script.)
	modifyDBPwd_yyyy-mm-dd.log	Run log file that records the DBService password change tool (You need to execute the **$DBSERVICE_HOME/sbin/modifyDBPwd.sh** script.)
	dbserver_switchover.log	Log for DBServer to execute the active/standby switchover script (the active/standby switchover needs to be performed)

Type	Log File Name	Description
GaussDB run log	gaussdb.log	Log file that records database running information
	gs_ctl-current.log	Log file that records operations performed by using the **gs_ctl** tool
	gs_guc-current.log	Log file that records operations, mainly parameter modification performed by using the **gs_guc** tool
	gaussdbinstall.log	GaussDB installation log file
	gaussdbuninstall.log	GaussDB uninstallation log file
HA script run log	floatip_ha.log	Log file that records the script of floating IP addresses
	gaussDB_ha.log	Log file that records the script of GaussDB resources
	ha_monitor.log	Log file that records the HA process monitoring information
	send_alarm.log	Alarm sending log file
	ha.log	HA run log file
DBService audit log	dbservice_audit.log	Audit log file that records DBService operations, such as backup and restoration operations

Log Format

The following table lists the DBService log formats.

Table 12-64 Log format

Type	Format	Example
Run log	[<*yyyy-MM-dd HH:mm:ss*>] <*Log level*>: [< *Name of the script that generates the log*: *Line number* >]: < *Message in the log*>	[2020-12-19 15:56:42] INFO [postinstall.sh:653] Is cloud flag is false. (main)
Audit log	[<*yyyy-MM-dd HH:mm:ss,SSS*>] UserName:<*Username*> UserIP:<*User IP address*> Operation:<*Operation content*> Result:<*Operation results*> Detail:<*Detailed information*>	[2020-05-26 22:00:23] UserName:omm UserIP:192.168.10.21 Operation:DBService data backup Result: SUCCESS Detail: DBService data backup is successful.

12.6 Using Flink

12.6.1 Using Flink from Scratch

This section describes how to use Flink to run wordcount jobs.

Prerequisites

- Flink has been installed in an MRS cluster.
- The cluster runs properly and the client has been correctly installed, for example, in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Using the Flink Client (Versions Earlier Than MRS 3.x)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to initialize environment variables:

source /opt/hadoopclient/bigdata_env

Step 4 If Kerberos authentication is enabled for the cluster, perform the following steps. If not, skip this whole step.

1. Prepare a user for submitting Flink jobs..
2. Log in to Manager and download the authentication credential.

   Log in to Manager of the cluster. For details, see **Accessing MRS Manager (Versions Earlier Than MRS 3.x)**. Choose **System Settings** > **User**

Management. In the **Operation** column of the row that contains the added user, choose **More** > **Download Authentication Credential**.

3. Decompress the downloaded authentication credential package and copy the **user.keytab** file to the client node, for example, to the **/opt/hadoopclient/Flink/flink/conf** directory on the client node. If the client is installed on a node outside the cluster, copy the **krb5.conf** file to the **/etc/** directory on this node.
4. Configure security authentication by adding the **keytab** path and username in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** configuration file.

   security.kerberos.login.keytab: *<user.keytab file path>*

   security.kerberos.login.principal: *<Username>*

   Example:

   security.kerberos.login.keytab: /opt/hadoopclient/Flink/flink/conf/user.keytab

   security.kerberos.login.principal: test
5. In the **bin** directory of the Flink client, run the following command to perform security hardening and set password to a new one for submitting jobs. For details, see "Using Flink" > "Security Hardening" > "Authentication and Encryption" in *MapReduce Service Component Operation Guide*.

   sh generate_keystore.sh <*password*>

   The script automatically replaces the SSL value in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** file. For an MRS 2.*x* or earlier security cluster, external SSL is disabled by default. To enable external SSL, configure the parameter and run the script again. For details, see "Using Flink" > "Security Hardening" in *MapReduce Service Component Operation Guide*.

   NOTE

   - You do not need to manually generate the **generate_keystore.sh** script.
   - After authentication and encryption, the generated **flink.keystore**, **flink.truststore**, and **security.cookie** items are automatically filled in the corresponding configuration items in **flink-conf.yaml**.
6. Configure paths for the client to access the **flink.keystore** and **flink.truststore** files.
   - Absolute path: After the script is executed, the file path of **flink.keystore** and **flink.truststore** is automatically set to the absolute path **/opt/hadoopclient/Flink/flink/conf/** in the **flink-conf.yaml** file. In this case, you need to move the **flink.keystore** and **flink.truststore** files from the **conf** directory to this absolute path on the Flink client and Yarn nodes.
   - Relative path: Perform the following steps to set the file path of **flink.keystore** and **flink.truststore** to the relative path and ensure that the directory where the Flink client command is executed can directly access the relative paths.
     i. Create a directory, for example, **ssl**, in **/opt/hadoopclient/Flink/flink/conf/**.

        cd /opt/hadoopclient/Flink/flink/conf/

        mkdir ssl

     ii. Move the **flink.keystore** and **flink.truststore** files to the **/opt/hadoopclient/Flink/flink/conf/ssl/** directory.

        mv flink.keystore ssl/

mv flink.truststore ssl/

iii. Change the values of the following parameters to relative paths in the **flink-conf.yaml** file:

```
security.ssl.internal.keystore: ssl/flink.keystore
security.ssl.internal.truststore: ssl/flink.truststore
```

Step 5 Run a wordcount job.

NOTICE

To submit or run jobs on Flink, the user must have the following permissions:

- If Ranger authentication is enabled, the current user must belong to the **hadoop** group or the user has been granted the **/flink** read and write permissions in Ranger.
- If Ranger authentication is disabled, the current user must belong to the **hadoop** group.

- Normal cluster (Kerberos authentication disabled)
  - Run the following commands to start a session and submit a job in the session:

    yarn-session.sh -nm "*session-name***"**

    flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - Run the following command to submit a single job on Yarn:

    flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
- Security cluster (Kerberos authentication enabled)
  - If the **flink.keystore** and **flink.truststore** file are stored in the absolute path:
    - Run the following commands to start a session and submit a job in the session:

      yarn-session.sh -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
    - Run the following command to submit a single job on Yarn:

      flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - If the **flink.keystore** and **flink.truststore** files are stored in the relative path:
    - In the same directory of SSL, run the following commands to start a session and submit jobs in the session. The SSL directory is a relative path. For example, if the SSL directory is **opt/hadoopclient/Flink/flink/conf/**, then run the following commands in this directory:

      yarn-session.sh -t ssl/ -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar

- Run the following command to submit a single job on Yarn:

  flink run -m yarn-cluster -yt ssl/ /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar

Step 6 After the job has been successfully submitted, the following information is displayed on the client:

Figure 12-6 Job submitted successfully on Yarn

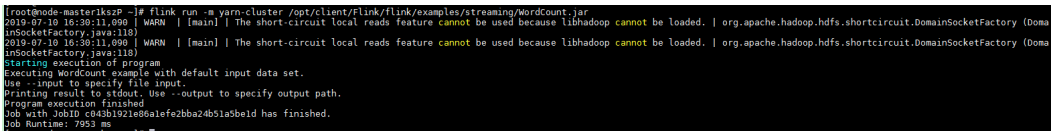
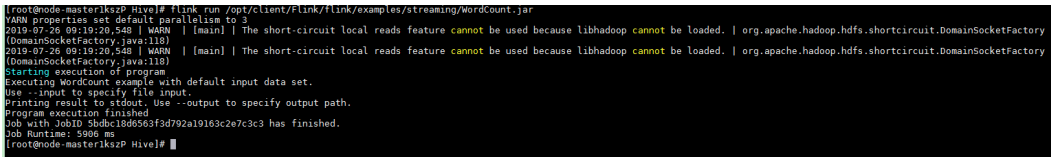


Figure 12-7 Session started successfully

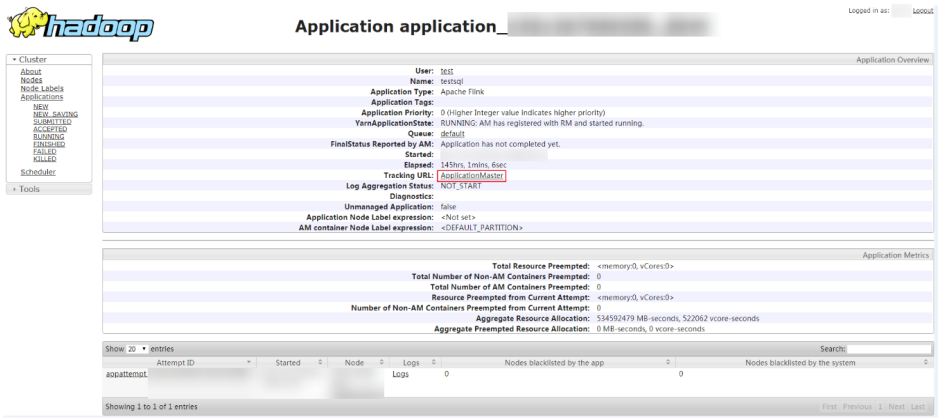
Figure 12-8 Job submitted successfully in the session



Step 7 Go to the native YARN service page, find the application of the job, and click the application name to go to the job details page. For details, see "Using Flink" > "Viewing Flink Job Information" in *MapReduce Service Component Operation Guide*.

- If the job is not completed, click **Tracking URL** to go to the native Flink page and view the job running information.
- If the job submitted in a session has been completed, you can click **Tracking URL** to log in to the native Flink service page to view job information.

Figure 12-9 Application



----End

Using the Flink Client (MRS 3.x or Later)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to initialize environment variables:

source /opt/hadoopclient/bigdata_env

Step 4 If Kerberos authentication is enabled for the cluster, perform the following steps. If not, skip this whole step.

1. Prepare a user for submitting Flink jobs.
2. Log in to Manager and download the authentication credential.

   Log in to Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **System** > **Permission** > **Manage User**. On the displayed page, locate the row that contains the added user, click **More** in the **Operation** column, and select **Download authentication credential**.
3. Decompress the downloaded authentication credential package and copy the **user.keytab** file to the client node, for example, to the **/opt/hadoopclient/Flink/flink/conf** directory on the client node. If the client is installed on a node outside the cluster, copy the **krb5.conf** file to the **/etc/** directory on this node.
4. Append the service IP address of the node where the client is installed, floating IP address of Manager, and IP address of the master node to the **jobmanager.web.access-control-allow-origin** and **jobmanager.web.allow-access-address** configuration item in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** file. Use commas (,) to separate IP addresses.

   jobmanager.web.access-control-allow-origin: *xx.xx.xxx.xxx,xx.xx.xxx.xxx,xx.xx.xxx.xxx*
   jobmanager.web.allow-access-address: *xx.xx.xxx.xxx,xx.xx.xxx.xxx,xx.xx.xxx.xxx*

NOTE

- To obtain the service IP address of the node where the client is installed, perform the following operations:
  - Node inside the cluster:

    In the navigation tree of the MRS management console, choose **Clusters > Active Clusters**, select a cluster, and click its name to switch to the cluster details page.

    On the **Nodes** tab page, view the IP address of the node where the client is installed.
  - Node outside the cluster: IP address of the ECS where the client is installed.
- To obtain the floating IP address of Manager, perform the following operations:
  - In the navigation tree of the MRS management console, choose **Clusters > Active Clusters**, select a cluster, and click its name to switch to the cluster details page.

    On the **Nodes** tab page, view the **Name**. The node that contains **master1** in its name is the Master1 node. The node that contains **master2** in its name is the Master2 node.
  - Log in to the Master2 node remotely, and run the **ifconfig** command. In the command output, **eth0:wsom** indicates the floating IP address of MRS Manager. Record the value of **inet**. If the floating IP address of MRS Manager cannot be queried on the Master2 node, switch to the Master1 node to query and record the floating IP address. If there is only one Master node, query and record the cluster manager IP address of the Master node.

5. Configure security authentication by adding the **keytab** path and username in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** configuration file.

   security.kerberos.login.keytab: *<user.keytab file path>*

   security.kerberos.login.principal: *<Username>*

   Example:

   security.kerberos.login.keytab: /opt/hadoopclient/Flink/flink/conf/user.keytab

   security.kerberos.login.principal: test

6. In the **bin** directory of the Flink client, run the following command to perform security hardening and set password to a new one for submitting jobs. For details, see "Using Flink" > "Security Hardening" > "Authentication and Encryption" in *MapReduce Service Component Operation Guide*.

   sh generate_keystore.sh <*password*>

   The script automatically replaces the SSL value in the **/opt/hadoopclient/Flink/flink/conf/flink-conf.yaml** file.

   sh generate_keystore.sh <*password*>

NOTE

After authentication and encryption, the **flink.keystore** and **flink.truststore** files are generated in the **conf** directory on the Flink client and the following configuration items are set to the default values in the **flink-conf.yaml** file:

- Set **security.ssl.keystore** to the absolute path of the **flink.keystore** file.
- Set **security.ssl.truststore** to the absolute path of the **flink.truststore** file.
- Set **security.cookie** to a random password automatically generated by the **generate_keystore.sh** script.
- By default, **security.ssl.encrypt.enabled** is set to **false** in the **flink-conf.yaml** file by default. The **generate_keystore.sh** script sets **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password** to the password entered when the **generate_keystore.sh** script is called.
- For MRS 3.1.0 or later, if ciphertext is required and **security.ssl.encrypt.enabled** is set to **true** in the **flink-conf.yaml** file, the **generate_keystore.sh** script does not set **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password**. To obtain the values, use the Manager plaintext encryption API by running the following command: **curl -k -i -u** *Username*:*Password* **-X POST -HContent-type:application/json -d '{"plainText":"***Password***"}' 'https://***x.x.x.x***:28443/web/api/v2/tools/encrypt'**

  In the preceding command, *Username*:*Password* indicates the user name and password for logging in to the system. The password of **"plainText"** indicates the one used to call the **generate_keystore.sh** script. *x.x.x.x* indicates the floating IP address of Manager.

7. Configure paths for the client to access the **flink.keystore** and **flink.truststore** files.
   - Absolute path: After the script is executed, the file path of **flink.keystore** and **flink.truststore** is automatically set to the absolute path **/opt/hadoopclient/Flink/flink/conf/** in the **flink-conf.yaml** file. In this case, you need to move the **flink.keystore** and **flink.truststore** files from the **conf** directory to this absolute path on the Flink client and Yarn nodes.
   - Relative path: Perform the following steps to set the file path of **flink.keystore** and **flink.truststore** to the relative path and ensure that the directory where the Flink client command is executed can directly access the relative paths.
     i. Create a directory, for example, **ssl**, in **/opt/hadoopclient/Flink/flink/conf/**.

        cd /opt/hadoopclient/Flink/flink/conf/

        mkdir ssl

     ii. Move the **flink.keystore** and **flink.truststore** files to the **/opt/hadoopclient/Flink/flink/conf/ssl/** directory.

        mv flink.keystore ssl/

        mv flink.truststore ssl/

     iii. Change the values of the following parameters to relative paths in the **flink-conf.yaml** file:

```
security.ssl.keystore: ssl/flink.keystore
security.ssl.truststore: ssl/flink.truststore
```

Step 5 Run a wordcount job.

NOTICE

To submit or run jobs on Flink, the user must have the following permissions:

- If Ranger authentication is enabled, the current user must belong to the **hadoop** group or the user has been granted the **/flink** read and write permissions in Ranger.
- If Ranger authentication is disabled, the current user must belong to the **hadoop** group.

- Normal cluster (Kerberos authentication disabled)
  - Run the following commands to start a session and submit a job in the session:

    yarn-session.sh -nm "*session-name***"**

    flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - Run the following command to submit a single job on Yarn:

    flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
- Security cluster (Kerberos authentication enabled)
  - If the **flink.keystore** and **flink.truststore** files are stored in the absolute path:
    - Run the following commands to start a session and submit a job in the session:

      yarn-session.sh -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
    - Run the following command to submit a single job on Yarn:

      flink run -m yarn-cluster /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
  - If the **flink.keystore** and **flink.truststore** file are stored in the relative path:
    - In the same directory of SSL, run the following commands to start a session and submit jobs in the session. The SSL directory is a relative path. For example, if the SSL directory is **opt/hadoopclient/Flink/flink/conf/**, then run the following commands in this directory:

      yarn-session.sh -t ssl/ -nm "*session-name***"**

      flink run /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar
    - Run the following command to submit a single job on Yarn:

      flink run -m yarn-cluster -yt ssl/ /opt/hadoopclient/Flink/flink/examples/streaming/WordCount.jar

Step 6 After the job has been successfully submitted, the following information is displayed on the client:

Figure 12-10 Job submitted successfully on Yarn

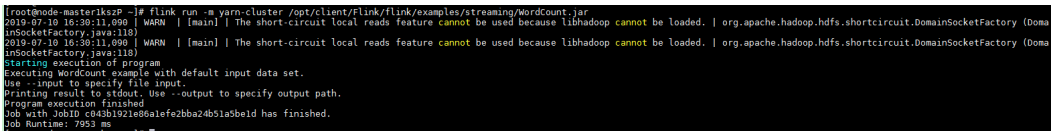


Figure 12-11 Session started successfully

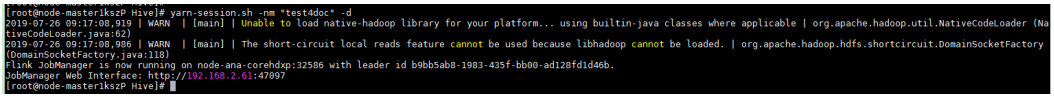
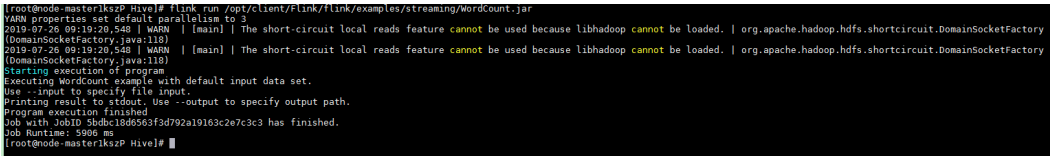


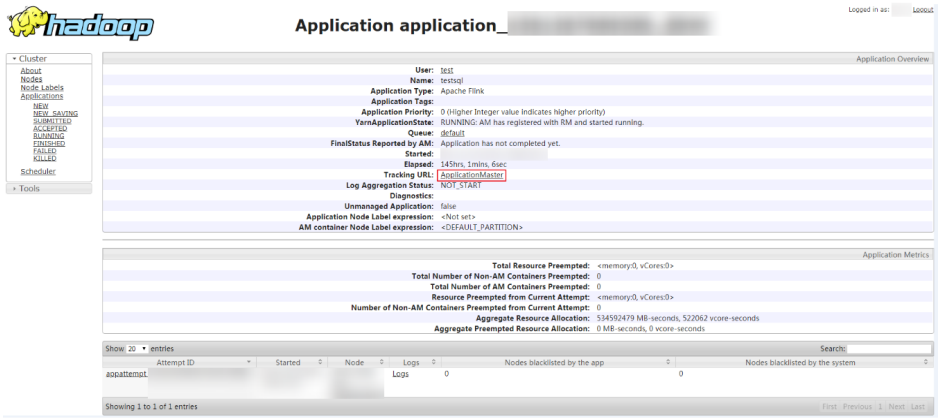
Figure 12-12 Job submitted successfully in the session



Step 7 Go to the native YARN service page, find the application of the job, and click the application name to go to the job details page. For details, see "Using Flink" > "Viewing Flink Job Information" in *MapReduce Service Component Operation Guide*.

- If the job is not completed, click **Tracking URL** to go to the native Flink page and view the job running information.
- If the job submitted in a session has been completed, you can click **Tracking URL** to log in to the native Flink service page to view job information.

Figure 12-13 Application



----End

12.6.2 Viewing Flink Job Information

You can view Flink job information on the Yarn web UI.

Prerequisites

The Flink service has been installed in a cluster.

Accessing the Yarn Web UI

Step 1 Go to the Yarn service page.

- For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **Yarn** > **Yarn Summary**.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Instance** > **Dashboard**.

Step 2 Click the link next to **ResourceManager WebUI** to go to the Yarn web UI page.

----End

12.6.3 Flink Configuration Management

12.6.3.1 Configuring Parameter Paths

All parameters of Flink must be set on a client. The path of a configuration file is as follows: *Client installation path*/**Flink/flink/conf/flink-conf.yaml**.

NOTE

- You are advised to set the parameters in the format of *Key*: *Value* in the **flink-conf.yaml** configuration file on the client.

  Example: **taskmanager.heap.size: 1024mb**

  A space is required between *Key*: and *Value*.
- If parameters are modified in the Flink service configuration, you need to download and install the client again after the configuration is complete.

12.6.3.2 JobManager & TaskManager

Scenarios

JobManager and TaskManager are main components of Flink. You can configure the parameters for different security and performance scenarios on the client.

Configuration Description

Main configuration items include communication port, memory management, connection retry, and so on.

For versions earlier than MRS 3.x, see **Table 12-65**.

Table 12-65 Parameters

Parameter	Mandatory	Default Value	Description
taskmanager.rpc.port	No	32326-32390	IPC port range of TaskManager
taskmanager.data.port	No	32391-32455	Data exchange port range of TaskManager
taskmanager.data.ssl.enabled	No	false	Whether to enable secure sockets layer (SSL) encryption for data transfer between TaskManagers. This parameter is valid only when the global switch **security.ssl** is enabled.
taskmanager.numberOfTaskSlots	No	3	Number of slots occupied by TaskManager. Generally, the value is configured as the number of cores of the physical machine. In **yarn-session** mode, the value can be transmitted by only the **-s** parameter. In **yarn-cluster** mode, the value can be transmitted by only the **-ys** parameter.
parallelism.default	No	1	Number of concurrent job operators.
taskmanager.memory.size	No	0	Amount of heap memory of the Java virtual machine (JVM) that TaskManager reserves for sorting, hash tables, and caching of intermediate results. If unspecified, the memory manager will take a fixed ratio with respect to the size of JVM as specified by **taskmanager.memory.fraction**. The unit is MB.
taskmanager.memory.fraction	No	0.7	Ratio of JVM heap memory that TaskManager reserves for sorting, hash tables, and caching of intermediate results.
taskmanager.memory.off-heap	Yes	false	Whether TaskManager uses off-heap memory for sorting, hash tables and intermediate status. You are advised to enable this item for large memory needs to improve memory operation efficiency.

Parameter	Mandatory	Default Value	Description
taskmanager.memory.segment-size	No	32768	Size of memory segment on TaskManager. Memory segment is the basic unit of the reserved memory space and is used to configure network buffer stacks. The unit is bytes.
taskmanager.memory.preallocate	No	false	Whether TaskManager allocates reserved memory space upon startup. You are advised to enable this item when off-heap memory is used.
taskmanager.registration.initial-backoff	No	500 ms	Initial interval between two consecutive registration attempts. The unit is ms/s/m/h/d. **NOTE** The time value and unit are separated by half-width spaces. ms/s/m/h/d indicates millisecond, second, minute, hour, and day, respectively.
taskmanager.registration.refused-backoff	No	5 min	Retry interval when a registration connection is rejected by JobManager.
task.cancellation.interval	No	30000	Interval between two successive task cancellation attempts.

For configuration items for MRS 3.x or later, see **Table 12-66**.

Table 12-66 Parameters

Parameter	Description	Default Value	Mandatory
taskmanager.rpc.port	IPC port range of TaskManager	32326-32390	No
client.rpc.port	Akka system listening port on the Flink client.	32651-32720	No
taskmanager.data.port	Data exchange port range of TaskManager	32391-32455	No

Parameter	Description	Default Value	Mandatory
taskmanager.data.ssl.enabled	Whether to enable secure sockets layer (SSL) encryption for data transfer between TaskManagers. This parameter is valid only when the global switch **security.ssl** is enabled.	false	No
jobmanager.heap.size	Size of the heap memory of JobManager. In **yarn-session** mode, the value can be transmitted by only the **-jm** parameter. In **yarn-cluster** mode, the value can be transmitted by only the **-yjm** parameter. If the value is smaller than **yarn.scheduler.minimum-allocation-mb** in the Yarn configuration file, the Yarn configuration value is used. Unit: B/KB/MB/GB/TB.	1024mb	No
taskmanager.heap.size	Size of the heap memory of TaskManager. In **yarn-session** mode, the value can be transmitted by only the **-tm** parameter. In **yarn-cluster** mode, the value can be transmitted by only the **-ytm** parameter. If the value is smaller than **yarn.scheduler.minimum-allocation-mb** in the Yarn configuration file, the Yarn configuration value is used. The unit is B/KB/MB/GB/TB.	1024mb	No
taskmanager.numberOfTaskSlots	Number of slots occupied by TaskManager. Generally, the value is configured as the number of cores of the physical machine. In **yarn-session** mode, the value can be transmitted by only the **-s** parameter. In **yarn-cluster** mode, the value can be transmitted by only the **-ys** parameter.	1	No
parallelism.default	Default degree of parallelism, which is used for jobs for which the degree of parallelism is not specified	1	No
taskmanager.network.numberOfBuffers	Number of TaskManager network transmission buffer stacks. If an error indicates insufficient system buffer, increase the parameter value.	2048	No
taskmanager.memory.fraction	Ratio of JVM heap memory that TaskManager reserves for sorting, hash tables, and caching of intermediate results.	0.7	No

Parameter	Description	Default Value	Mandatory
taskmanager.memory.off-heap	Whether TaskManager uses off-heap memory for sorting, hash tables and intermediate status. You are advised to enable this item for large memory needs to improve memory operation efficiency.	false	Yes
taskmanager.memory.segment-size	Size of the memory buffer used by the memory manager and network stack The unit is bytes.	32768	No
taskmanager.memory.preallocate	Whether TaskManager allocates reserved memory space upon startup. You are advised to enable this item when off-heap memory is used.	false	No
taskmanager.debug.memory.startLogThread	Enable this item for debugging Flink memory and garbage collection (GC)-related problems. TaskManager periodically collects memory and GC statistics, including the current utilization of heap and off-heap memory pools and GC time.	false	No
taskmanager.debug.memory.logIntervalMs	Interval at which TaskManager periodically collects memory and GC statistics.	0	No
taskmanager.maxRegistrationDuration	Maximum duration of TaskManager registration on JobManager. If the actual duration exceeds the value, TaskManager is disabled.	5 min	No
taskmanager.initial-registration-pause	Initial interval between two consecutive registration attempts. The value must contain a time unit (ms/s/min/h/d), for example, 5 seconds.	500ms **NOTE** The time value and unit are separated by half-width spaces. ms/s/m/h/d indicates millisecond, second, minute, hour, and day, respectively.	No

Parameter	Description	Default Value	Mandatory
taskmanager.max-registration-pause	Maximum registration retry interval in case of TaskManager registration failures. The unit is ms/s/m/h/d.	30s	No
taskmanager.refused-registration-pause	Retry interval when a TaskManager registration connection is rejected by JobManager. The unit is ms/s/m/h/d.	10s	No
task.cancellation.interval	Interval between two successive task cancellation attempts. The unit is millisecond.	30000	No
classloader.resolve-order	Class resolution policies defined when classes are loaded from user codes, which means whether to first check the user code JAR file (**child-first**) or the application class path (**parent-first**). The default setting indicates that the class is first loaded from the user code JAR file, which means that the user code JAR file can contain and load dependencies that are different from those used by Flink.	child-first	No
slot.idle.timeout	Timeout for an idle slot in Slot Pool, in milliseconds.	50000	No
slot.request.timeout	Timeout for requesting a slot from Slot Pool, in milliseconds.	300000	No
task.cancellation.timeout	Timeout of task cancellation, in milliseconds. If a task cancellation times out, a fatal TaskManager error may occur. If this parameter is set to **0**, no error is reported when a task cancellation times out.	180000	No
taskmanager.network.detailed-metrics	Indicates whether to enable the detailed metrics monitoring of network queue lengths.	false	No

Parameter	Description	Default Value	Mandatory
taskmanager.network.memory.buffers-per-channel	Maximum number of network buffers used by each output/input channel (sub-partition/incoming channel). In credit-based flow control mode, this indicates how much credit is in each input channel. It should be configured with at least 2 buffers to deliver good performance. One buffer is used to receive in-flight data in the sub-partition, and the other for parallel serialization.	2	No
taskmanager.network.memory.floating-buffers-per-gate	Number of extra network buffers used by each output gate (result partition) or input gate, indicating the amount of floating credit shared among all input channels in credit-based flow control mode. Floating buffers are distributed based on the backlog feedback (real-time output buffers in sub-partitions) and can help mitigate back pressure caused by unbalanced data distribution among sub-partitions. Increase this value if the round-trip time between nodes is long and/or the number of machines in the cluster is large.	8	No
taskmanager.network.memory.fraction	Ratio of JVM memory used for network buffers, which determines how many streaming data exchange channels a TaskManager can have at the same time and the extent of channel buffering. Increase this value or the values of **taskmanager.network.memory.min** and **taskmanager.network.memory.max** if the job is rejected or a warning indicating that the system does not have enough buffers is received. Note that the values of **taskmanager.network.memory.min** and **taskmanager.network.memory.max** may overwrite this value.	0.1	No
taskmanager.network.memory.max	Maximum memory size of the network buffer. The value must contain a unit (B/KB/MB/GB/TB).	1 GB	No
taskmanager.network.memory.min	Minimum memory size of the network buffer. The value must contain a unit (B/KB/MB/GB/TB).	64 MB	No

Parameter	Description	Default Value	Mandatory
taskmanager.network.request-backoff.initial	Minimum backoff for partition requests of input channels.	100	No
taskmanager.network.request-backoff.max	Maximum backoff for partition requests of input channels.	10000	No
taskmanager.registration.timeout	Timeout for TaskManager registration. TaskManager will be terminated if it is not successfully registered within the specified time. The value must contain a time unit (ms/s/min/h/d).	5 min	No
resourcemanager.taskmanager-timeout	Timeout interval for releasing an idle TaskManager, in milliseconds.	30000	No

12.6.3.3 Blob

Scenarios

The Blob server on the JobManager node is used to receive JAR files uploaded by users on the client, send JAR files to TaskManager, and transfer log files. Flink provides some items for configuring the Blob server. You can configure them in the **flink-conf.yaml** configuration file.

Configuration Description

Users can configure the port, SSL, retry times, and concurrency.

Table 12-67 Parameters

Parameter	Description	Default Value	Mandatory
blob.server.port	Blob server port	32456 to 32520	No
blob.service.ssl.enabled	Indicates whether to enable the encryption for the blob transmission channel. This parameter is valid only when the global switch **security.ssl** is enabled.	true	Yes

Parameter	Description	Default Value	Mandatory
blob.fetch.retries	Number of times that TaskManager tries to download blob files from JobManager.	50	No
blob.fetch.num-concurrent	Number of concurrent tasks for downloading blob files supported by JobManager.	50	No
blob.fetch.backlog	Number of blob files, such as **.jar** files, to be downloaded in the queue supported by JobManager. The unit is count.	1000	No
library-cache-manager.cleanup.interval	Interval at which JobManager deletes the JAR files stored on the HDFS when the user cancels the Flink job. The unit is second.	3600	No

NOTE

For versions earlier than MRS 3.x, **library-cache-manager.cleanup.interval** cannot be configured.

12.6.3.4 Distributed Coordination (via Akka)

Scenarios

The Akka actor model is the basis of communications between the Flink client and JobManager, JobManager and TaskManager, as well as TaskManager and TaskManager. Flink enables you to configure the Akka connection parameters in the **flink-conf.yaml** file based on the network environment or optimization policy.

Configuration Description

You can configure timeout settings of message sending and waiting, and the Akka listening mechanism Deathwatch.

For versions earlier than MRS 3.x, see **Table 12-68**.

Table 12-68 Parameters

Parameter	Mandatory	Default Value	Description
akka.ask.timeout	No	10 s	Timeout duration of Akka asynchronous and block requests. If a Flink timeout failure occurs, this value can be increased. Timeout occurs when the machine processing speed is slow or the network is blocked. The unit is ms/s/m/h/d.
akka.lookup.timeout	No	10 s	Timeout duration for JobManager actor object searching. The unit is ms/s/m/h/d.
akka.framesize	No	10485760b	Maximum size of the message transmitted between JobManager and TaskManager. If a Flink error occurs because the message exceeds this limit, the value can be increased. The unit is b/B/KB/MB.
akka.watch.heartbeat.interval	No	10 s	Heartbeat interval at which the Akka DeathWatch mechanism detects disconnected TaskManager. If TaskManager is frequently and incorrectly marked as disconnected due to heartbeat loss or delay, the value can be increased. The unit is ms/s/m/h/d. **NOTE** For detailed description of Akka DeathWatch, see the Akka official website: **http://doc.akka.io/docs/akka/snapshot/scala/remoting.html#failure-detector**.
akka.watch.heartbeat.pause	No	60 s	Acceptable heartbeat pause for Akka DeathWatch mechanism. A small value indicates that irregular heartbeat is not accepted. The unit is ms/s/m/h/d. **NOTE** For detailed description of Akka DeathWatch, see the Akka official website: **http://doc.akka.io/docs/akka/snapshot/scala/remoting.html#failure-detector**.
akka.watch.threshold	No	12	DeathWatch failure detection threshold. A small value is prone to mark normal TaskManager as failed and a large value increases failure detection time. **NOTE** For detailed description of Akka DeathWatch, see the Akka official website: **http://doc.akka.io/docs/akka/snapshot/scala/remoting.html#failure-detector**.

Parameter	Mandatory	Default Value	Description
akka.tcp.timeout	No	20 s	Timeout duration of Transmission Control Protocol (TCP) connection request. If TaskManager connection timeout occurs frequently due to the network congestion, the value can be increased. The unit is ms/s/m/h/d.
akka.throughput	No	15	Number of messages processed by Akka in batches. After an operation, the processing thread is returned to the thread pool. A small value indicates the fair scheduling for actor message processing. A large value indicates improved overall performance but lowered scheduling fairness.
akka.log.lifecycle.events	No	false	Switch of Akka remote time logging, which can be enabled for debugging.
akka.startup-timeout	No	The default value is the same as the value of **akka.ask.timeout**.	Timeout duration of remote component started by Akka. The unit is ms/s/m/h/d.
akka.ssl.enabled	Yes	true	Switch of Akka communication SSL. This parameter is valid only when the global switch **security.ssl** is enabled.

For configuration items for MRS 3.x or later, see **Table 12-69**.

Table 12-69 Parameters

Parameter	Description	Default Value	Mandatory
akka.ask.timeout	Timeout duration of Akka asynchronous and block requests. If a Flink timeout failure occurs, this value can be increased. Timeout occurs when the machine processing speed is slow or the network is blocked. The unit is ms/s/m/h/d.	10s	No

Parameter	Description	Default Value	Mandatory
akka.lookup.timeout	Timeout duration for JobManager actor object searching. The unit is ms/s/m/h/d.	10s	No
akka.framesize	Maximum size of the message transmitted between JobManager and TaskManager. If a Flink error occurs because the message exceeds this limit, the value can be increased. The unit is b/B/KB/MB.	10485760b	No
akka.watch.heartbeat.interval	Heartbeat interval at which the Akka DeathWatch mechanism detects disconnected TaskManager. If TaskManager is frequently and incorrectly marked as disconnected due to heartbeat loss or delay, the value can be increased. The unit is ms/s/m/h/d. **NOTE** For detailed explanation of DeathWatch, see the Akka official website: **http://doc.akka.io/docs/akka/snapshot/scala/remoting.html#failure-detector**.	10s	No
akka.watch.heartbeat.pause	Acceptable heartbeat pause for Akka DeathWatch mechanism. A small value indicates that irregular heartbeat is not accepted. The unit is ms/s/m/h/d. **NOTE** For detailed explanation of DeathWatch, see the Akka official website: **http://doc.akka.io/docs/akka/snapshot/scala/remoting.html#failure-detector**.	60s	No
akka.watch.threshold	DeathWatch failure detection threshold. A small value may mark normal TaskManager as failed and a large value increases failure detection time. **NOTE** For detailed explanation of DeathWatch, see the Akka official website: **http://doc.akka.io/docs/akka/snapshot/scala/remoting.html#failure-detector**.	12	No

Parameter	Description	Default Value	Mandatory
akka.tcp.timeout	Timeout duration of Transmission Control Protocol (TCP) connection request. If TaskManager connection timeout occurs frequently due to the network congestion, the value can be increased. The unit is ms/s/m/h/d.	20s	No
akka.throughput	Number of messages processed by Akka in batches. After an operation, the processing thread is returned to the thread pool. A small value indicates the fair scheduling for actor message processing. A large value indicates improved overall performance but lowered scheduling fairness.	15	No
akka.log.lifecycle.events	Switch of Akka remote time logging, which can be enabled for debugging.	false	No
akka.startup-timeout	Timeout interval before a remote component fails to be started. The value must contain a time unit (ms/s/min/h/d).	The default value is the same as the value of **akka.ask.timeout**.	No
akka.ssl.enabled	Switch of Akka communication SSL. This parameter is valid only when the global switch **security.ssl** is enabled.	true	Yes
akka.client-socket-worker-pool.pool-size-factor	Factor that is used to determine the thread pool size. The pool size is calculated based on the following formula: ceil (available processors * factor). The size is bounded by the **pool-size-min** and **pool-size-max** values.	1.0	No
akka.client-socket-worker-pool.pool-size-max	Maximum number of threads calculated based on the factor.	2	No
akka.client-socket-worker-pool.pool-size-min	Minimum number of threads calculated based on the factor.	1	No

Parameter	Description	Default Value	Mandatory
akka.client.timeout	Timeout duration of the client. The value must contain a time unit (ms/s/min/h/d).	60s	No
akka.server-socket-worker-pool.pool-size-factor	Factor that is used to determine the thread pool size. The pool size is calculated based on the following formula: ceil (available processors * factor). The size is bounded by the **pool-size-min** and **pool-size-max** values.	1.0	No
akka.server-socket-worker-pool.pool-size-max	Maximum number of threads calculated based on the factor.	2	No
akka.server-socket-worker-pool.pool-size-min	Minimum number of threads calculated based on the factor.	1	No

12.6.3.5 SSL

Scenarios

When the secure Flink cluster is required, SSL-related configuration items must be set.

Configuration Description

Configuration items include the SSL switch, certificate, password, and encryption algorithm.

For versions earlier than MRS 3.x, see **Table 12-70**.

Table 12-70 Parameters

Parameter	Mandatory	Default Value	Description
security.ssl.internal.enabled	Yes	The value is automatically configured according to the cluster installation mode. • Security mode: The default value is **true**. • Normal mode: The default value is **false**.	Main switch of internal communication SSL.
security.ssl.internal.keystore	Yes	-	Java keystore file.
security.ssl.internal.keystore-password	Yes	-	Password used to decrypt the keystore file.
security.ssl.internal.key-password	Yes	-	Password used to decrypt the server key in the keystore file.
security.ssl.internal.truststore	Yes	-	**truststore** file containing the public CA certificates.
security.ssl.internal.truststore-password	Yes	-	Password used to decrypt the truststore file.
security.ssl.protocol	Yes	TLSv1.2	SSL transmission protocol version
security.ssl.algorithms	Yes	The default value is **TLS_RSA_WITH_AES_128_CBC_SHA256,TLS_DHE_RSA_WITH_AES_128_CBC_SHA256,TLS_DHE_DSS_WITH_AES_128_CBC_SHA256**.	Supported SSL standard algorithm. For details, see the Java official website: **http://docs.oracle.com/javase/8/docs/technotes/guides/security/StandardNames.html#ciphersuites**.

Parameter	Mandatory	Default Value	Description
security.ssl.rest.enabled	Yes	The value is automatically configured according to the cluster installation mode. • Security mode: The default value is **true**. • Normal mode: The default value is **false**.	Main switch of external communication SSL.
security.ssl.rest.keystore	Yes	-	Java keystore file.
security.ssl.rest.keystore-password	Yes	-	Password used to decrypt the keystore file.
security.ssl.rest.key-password	Yes	-	Password used to decrypt the server key in the keystore file.
security.ssl.rest.truststore	Yes	-	**truststore** file containing the public CA certificates.
security.ssl.rest.truststore-password	Yes	-	Password used to decrypt the truststore file.

For configuration items for MRS 3.x or later, see **Table 12-71**.

Table 12-71 Parameters

Parameter	Description	Default Value	Mandatory
security.ssl.enabled	Main switch of internal communication SSL.	The value is automatically configured according to the cluster installation mode. • Security mode: The default value is **true**. • Non-security mode: The default value is **false**.	Yes
security.ssl.keystore	Java keystore file.	-	Yes

Parameter	Description	Default Value	Mandatory
security.ssl.keystore-password	Password used to decrypt the keystore file.	-	Yes
security.ssl.key-password	Password used to decrypt the server key in the keystore file.	-	Yes
security.ssl.truststore	**truststore** file containing the public CA certificates.	-	Yes
security.ssl.truststore-password	Password used to decrypt the truststore file.	-	Yes
security.ssl.protocol	SSL transmission protocol version.	TLSv1.2	Yes
security.ssl.algorithms	Supported SSL standard algorithm. For details, see the Java official website: **http://docs.oracle.com/javase/8/docs/technotes/guides/security/StandardNames.html#ciphersuites**.	The default value: "TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384"	Yes

12.6.3.6 Network communication (via Netty)

Scenario

When Flink runs a job, data transmission and reverse pressure detection between tasks depend on Netty. In certain environments, **Netty** parameters should be configured.

Configuration Description

For advanced optimization, you can modify the following Netty configuration items. The default configuration can meet the requirements of tasks of large-scale clusters with high concurrent throughput. For details about the parameters, visit the Netty official website at **http://netty.io/**.

Table 12-72 Parameter description

Parameter	Description	Default Value	Mandatory
taskmanager.network.netty.num-arenas	Number of Netty memory blocks.	1	No
taskmanager.network.netty.server.numThreads	Number of Netty server threads	1	No
taskmanager.network.netty.client.numThreads	Number of Netty client threads	1	No
taskmanager.network.netty.client.connectTimeoutSec	Netty client connection timeout duration. Unit: second	120	No
taskmanager.network.netty.sendReceiveBufferSize	Size of Netty sending and receiving buffers. This defaults to the system buffer size (**cat /proc/sys/net/ipv4/tcp_[rw]mem**) and is 4 MB in modern Linux. Unit: byte	4096	No
taskmanager.network.netty.transport	Netty transport type, either **nio** or **epoll**	nio	No

12.6.3.7 JobManager Web Frontend

Scenarios

When JobManager is started, the web server in the same process is also started.

- You can access the web server to obtain information about the current Flink cluster, including information about JobManager, TaskManager, and running jobs in the cluster.
- You can configure parameters of the web server.

Configuration Description

Configuration items include the port, temporary directory, display items, error redirection, and security-related items.

For versions earlier than MRS 3.x, see **Table 12-73**.

Table 12-73 Parameters

Parameter	Mandatory	Default Value	Description
jobmanager.web.port	No	32261-32325	Web port. Value range: 32261-32325.
jobmanager.web.allow-access-address	Yes	*	Web access whitelist. IP addresses are separated by commas (,). Only IP addresses in the whitelist can access the web.

For details about configuration items of MRS 3.*x* or later, see **Table 12-74**.

Table 12-74 Parameters

Parameter	Description	Default Value	Mandatory
flink.security.enable	When installing a Flink cluster, you are required to select **security mode** or **normal mode**. • If **security mode** is selected, the value of **flink.security.enable** is automatically set to **true**. • If **normal mode** is selected, the value of **flink.security.enable** is automatically set to **false**. If you want to checker whether Flink cluster is in security mode or normal mode, view the value of **flink.security.enable**.	The value is automatically configured based on the cluster installation mode.	No
rest.bind-port	Web port. Value range: 32261-32325.	32261-32325	No
jobmanager.web.history	Number of recent jobs to be displayed.	5	No
jobmanager.web.checkpoints.disable	Indicates whether to disable checkpoint statistics.	false	No
jobmanager.web.checkpoints.history	Number of checkpoint statistical records.	10	No
jobmanager.web.backpressure.cleanup-interval	Interval for clearing unaccessed backpressure records. The unit is millisecond.	600000	No

Parameter	Description	Default Value	Mandatory
jobmanager.web.backpressure.refresh-interval	Interval for updating backpressure records. The unit is millisecond.	60000	No
jobmanager.web.backpressure.num-samples	Number of stack tracing records for reverse pressure calculation.	100	No
jobmanager.web.backpressure.delay-between-samples	Sampling interval for reverse pressure calculation. The unit is millisecond.	50	No
jobmanager.web.ssl.enabled	Whether SSL encryption is enabled for web transmission. This parameter is valid only when the global switch **security.ssl** is enabled.	false	Yes
jobmanager.web.accesslog.enable	Switch to enable or disable web operation logs. The log is stored in **webaccess.log**.	true	Yes
jobmanager.web.x-frame-options	Value of the HTTP security header **X-Frame-Options**. The value can be **SAMEORIGIN**, **DENY**, or **ALLOW-FROM uri**.	DENY	Yes
jobmanager.web.cache-directive	Whether the web page can be cached.	no-store	Yes
jobmanager.web.expires-time	Expiration duration of web page cache. The unit is millisecond.	0	Yes
jobmanager.web.allow-access-address	Web access whitelist. IP addresses are separated by commas (,). Only IP addresses in the whitelist can access the web.	*	Yes
jobmanager.web.access-control-allow-origin	Web page same-origin policy that prevents cross-domain attacks.	*	Yes
jobmanager.web.refresh-interval	Web page refresh interval. The unit is millisecond.	3000	Yes
jobmanager.web.logout-timer	Automatic logout interval when no operation is performed. The unit is millisecond.	600000	Yes
jobmanager.web.403-redirect-url	Web page access error 403. If 403 error occurs, the page switch to a specified page.	Automatic configuration	Yes

Parameter	Description	Default Value	Mandatory
jobmanager.web.404-redirect-url	Web page access error 404. If 404 error occurs, the page switch to a specified page.	Automatic configuration	Yes
jobmanager.web.415-redirect-url	Web page access error 415. If 415 error occurs, the page switch to a specified page.	Automatic configuration	Yes
jobmanager.web.500-redirect-url	Web page access error 500. If 500 error occurs, the page switch to a specified page.	Automatic configuration	Yes
rest.await-leader-timeout	Time of the client waiting for the leader address. The unit is millisecond.	30000	No
rest.client.max-content-length	Maximum content length that the client handles (unit: bytes).	104857600	No
rest.connection-timeout	Maximum time for the client to establish a TCP connection (unit: ms).	15000	No
rest.idleness-timeout	Maximum time for a connection to stay idle before failing (unit: ms).	300000	No
rest.retry.delay	The time that the client waits between retries (unit: ms).	3000	No
rest.retry.max-attempts	The number of retry times if a retrievable operator fails.	20	No
rest.server.max-content-length	Maximum content length that the server handles (unit: bytes).	104857600	No
rest.server.numThreads	Maximum number of threads for the asynchronous processing of requests.	4	No
web.timeout	Timeout for web monitor (unit: ms).	10000	No

12.6.3.8 File Systems

Scenario

Result files are created when tasks are running. Flink enables you to configure parameters for file creation.

Configuration Description

Configuration items include overwriting policy and directory creation.

Table 12-75 Parameter description

Parameter	Description	Default Value	Mandatory
fs.overwrite-files	Whether to overwrite the existing file by default when the file is written.	false	No
fs.output.always-create-directory	When the degree of parallelism (DOP) of file writing programs is greater than 1, a directory is created under the output file path and different result files (one for each parallel writing program) are stored in the directory. • If this parameter is set to **true**, a directory is created for the writing program whose DOP is 1 and a result file is stored in the directory. • If this parameter is set to **false**, the file of the writing program whose DOP is 1 is created directly in the output path and no directory is created.	false	No

12.6.3.9 State Backend

Scenarios

Flink enables HA and job exception, as well as job pause and recovery during version upgrade. Flink depends on state backend to store job states and on the restart strategy to restart a job. You can configure state backend and the restart strategy.

Configuration Description

Configuration items include the state backend type, storage path, and restart strategy.

Table 12-76 Parameters

Parameter	Description	Default Value	Mandatory
state.backend.fs.checkpointdir	Path when the backend is set to **filesystem**. The path must be accessible by JobManager. Only the local mode is supported. In the cluster mode, use an HDFS path.	hdfs:///flink/checkpoints	No
state.savepoints.dir	Savepoint storage directory used by Flink to restore and update jobs. When a savepoint is triggered, the metadata of the savepoint is saved to this directory.	hdfs:///flink/savepoint	Mandatory in security mode
restart-strategy	Default restart policy, which is used for jobs for which no restart policy is specified. The options are as follows: • fixed-delay • failure-rate • none	none	No
restart-strategy.fixed-delay.attempts	Number of retry times when the fixed-delay restart strategy is used. For details, see **https://ci.apache.org/projects/flink/flink-docs-release-1.12/dev/task_failure_recovery.html**.	• If the checkpoint is enabled, the default value is the value of **Integer.MAX_VALUE**. • If the checkpoint is disabled, the default value is 3.	No

Parameter	Description	Default Value	Mandatory
restart-strategy.fixed-delay.delay	Retry interval when the fixed-delay strategy is used. The unit is ms/s/m/h/d.	• If the checkpoint is enabled, the default value is 10s. • If the checkpoint is disabled, the default value is the value of **akka.ask.timeout**.	No
restart-strategy.failure-rate.max-failures-per-interval	Maximum number of restart times in a specified period before a job fails when the fault rate policy is used. For details about the policies, see **https://ci.apache.org/projects/flink/flink-docs-release-1.12/dev/task_failure_recovery.html**.	1	No
restart-strategy.failure-rate.failure-rate-interval	Retry interval when the failure-rate strategy is used. The unit is ms/s/m/h/d.	60 s	No
restart-strategy.failure-rate.delay	Retry interval when the failure-rate strategy is used. The unit is ms/s/m/h/d.	The default value is the same as the value of **akka.ask.timeout**. For details, see **Distributed Coordination (via Akka)**.	No

12.6.3.10 Kerberos-based Security

Scenarios

Flink Kerberos configuration items must be configured in security mode.

Configuration Description

The configuration items include **keytab**, **principal**, and **cookie** of Kerberos.

NOTE

For versions earlier than MRS 3.x, the configuration item does not contain cookie.

Table 12-77 Parameters

Parameter	Description	Default Value	Mandatory
security.kerberos.login.keytab	Keytab file path. This parameter is a client parameter.	Configure the parameter based on actual service requirements.	Yes
security.kerberos.login.principal	A parameter on the client. If **security.kerberos.login.keytab** and **security.kerberos.login.principal** are both set, keytab certificate is used by default.	Configure the parameter based on actual service requirements.	No
security.kerberos.login.contexts	Contexts of the jass file generated by Flink. This parameter is a server parameter.	Client, KafkaClient	Yes
security.enable	Certificate enabling switch of the Flink internal module. This parameter is a client parameter.	This parameter is configured automatically according to the cluster installation mode. • Security mode: The default value is **true**. • Non-security mode: The default value is **false**.	Yes
security.cookie	Module certificate token. This parameter is a client parameter. It must be configured and cannot be left empty when **security.enable** is enabled.	Configure the parameter based on actual service requirements.	Yes

NOTE

For versions earlier than MRS 3.x, the configuration parameters do not include **security.enable** and **security.cookie**.

12.6.3.11 HA

Scenarios

The Flink HA mode depends on ZooKeeper. Therefore, ZooKeeper-related configuration items must be set.

Configuration Description

Configuration items include the ZooKeeper address, path, and security certificate.

Table 12-78 Parameters

Parameter	Description	Default Value	Mandatory
high-availability	Whether HA is enabled. Only the following two modes are supported currently: 1. none: Only a single JobManager is running. The checkpoint is disabled for JobManager. 2. ZooKeeper: • In non-Yarn mode, multiple JobManagers are supported and the leader JobManager is elected. • In Yarn mode, only one JobManager exists.	zookeeper	No
high-availability.zookeeper.quorum	ZooKeeper quorum address.	Automatic configuration	No
high-availability.zookeeper.path.root	Root directory that Flink creates on ZooKeeper, storing metadata required in HA mode.	/flink	No
high-availability.storageDir	Directory for storing JobManager metadata of state backend. ZooKeeper stores only pointers to actual data.	hdfs:///flink/recovery	No

Parameter	Description	Default Value	Mandatory
high-availability.zookeeper.client.session-timeout	Session timeout duration on the ZooKeeper client. The unit is millisecond.	60000	No
high-availability.zookeeper.client.connection-timeout	Connection timeout duration on the ZooKeeper client. The unit is millisecond.	15000	No
high-availability.zookeeper.client.retry-wait	Retry waiting time on the ZooKeeper client. The unit is millisecond.	5000	No
high-availability.zookeeper.client.max-retry-attempts	Maximum retry times on the ZooKeeper client.	3	No
high-availability.job.delay	Delay of job restart when JobManager recovers.	The default value is the same as the value of **akka.ask.timeout**.	No
high-availability.zookeeper.client.acl	ACL (open creator) of the ZooKeeper node. For ACL options, see **https://zookeeper.apache.org/doc/r3.5.1-alpha/zookeeperProgrammers.html#sc_BuiltinACLSchemes**.	This parameter is configured automatically according to the cluster installation mode. • Security mode: The default value is **creator**. • Non-security mode: The default value is **open**.	Yes

Parameter	Description	Default Value	Mandatory
zookeeper.sasl.disable	Simple authentication and security layer (SASL)-based certificate enable switch.	This parameter is configured automatically according to the cluster installation mode. • Security mode: The default value is **false**. • Non-security mode: The default value is **true**.	Yes
zookeeper.sasl.service-name	• If the ZooKeeper server configures a service whose name is different from **ZooKeeper**, this configuration item can be set. • If service names on the client and server are inconsistent, authentication fails.	zookeeper	Yes

NOTE

For versions earlier than MRS 3.x, the **high-availability.job.delay** parameter is not supported.

12.6.3.12 Environment

Scenario

In scenarios raising special requirements on JVM configuration, users can use configuration items to transfer JVM parameters to the client, JobManager, and TaskManager.

Configuration

Configuration items include JVM parameters.

Table 12-79 Parameter description

Parameter	Description	Default Value	Mandatory
env.java.opts	JVM parameter, which is transferred to the startup script, JobManager, TaskManager, and Yarn client. For example, transfer remote debugging parameters.	-Xloggc:<LOG_DIR>/gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=20M -Djdk.tls.ephemeralDHKeySize=2048 -Djava.library.path=${HADOOP_COMMON_HOME}/lib/native -Djava.net.preferIPv4Stack=true -Djava.net.preferIPv6Addresses=false -Dbeetle.application.home.path=/opt/*xxx*/Bigdata/common/runtime/security/config	No

12.6.3.13 Yarn

Scenario

Flink runs on a Yarn cluster and JobManager runs on ApplicationMaster. Certain configuration parameters of JobManager depend on Yarn. By setting Yarn-related configuration items, Flink is enabled to run better on Yarn.

Configuration Description

The configuration items include the memory, virtual kernel, and port of the Yarn container.

Table 12-80 Parameter description

Parameter	Description	Default Value	Mandatory
yarn.maximum-failed-containers	Maximum number of containers the system is going to reallocate in case of a container failure of TaskManager The default value is the number of TaskManagers when the Flink cluster is started.	5	No

Parameter	Description	Default Value	Mandatory
yarn.application-attempts	Number of ApplicationMaster restarts. The value is the maximum value in the validity interval that is set to Akka's timeout in Flink. After the restart, the IP address and port number of ApplicationMaster will change and you will need to connect to the client manually.	2	No
yarn.heartbeat-delay	Time between heartbeats with the ApplicationMaster and Yarn ResourceManager in seconds. Unit: second	5	No
yarn.containers.vcores	Number of virtual cores of each Yarn container	The default value is the number of TaskManager slots.	No
yarn.application-master.port	ApplicationMaster port number setting. A port number range is supported.	32586-32650	No

12.6.3.14 Pipeline

Scenarios

The Netty connection is used among multiple jobs to reduce latency. In this case, NettySink is used on the server and NettySource is used on the client for data transmission.

This section applies to MRS 3.x or later.

Configuration Description

Configuration items include NettySink information storing path, range of NettySink listening port, whether to enable SSL encryption, domain of the network used for NettySink monitoring.

Table 12-81 Parameters

Parameter	Description	Default Value	Mandatory
nettyconnector.registerserver.topic.storage	Path (on a third-party server) to information about IP address, port numbers, and concurrency of NettySink. ZooKeeper is recommended for storage.	/flink/nettyconnector	No. However, if pipeline is enabled, the feature is mandatory.
nettyconnector.sinkserver.port.range	Port range of NettySink.	If MRS cluster is used, the default value is 28444-28843.	No. However, if pipeline is enabled, the feature is mandatory.
nettyconnector.ssl.enabled	Whether SSL encryption for the communication between NettySink and NettySource is enabled. For details about the encryption key and protocol, see **SSL**.	false	No. However, if pipeline is enabled, the feature is mandatory.
nettyconnector.message.delimiter	Delimiter used to configure the message sent by NettySink to the NettySource, which is 2-4 bytes long, and cannot contain **\n**, **#**, or space.	The default value is **$_**.	No. However, if pipeline is enabled, the feature is mandatory.

12.6.4 Security Configuration

12.6.4.1 Security Features

Security Features of Flink

- All Flink cluster components support authentication.
  - The Kerberos authentication is supported between Flink cluster components and external components, such as Yarn, HDFS, and ZooKeeper.
  - The security cookie authentication between Flink cluster components, for example, Flink client and JobManager, JobManager and TaskManager, and TaskManager and TaskManager, are supported.
- SSL encrypted transmission is supported by Flink cluster components.
- SSL encrypted transmission between Flink cluster components, for example, Flink client and JobManager, JobManager and TaskManager, and TaskManager and TaskManager, are supported.
- Following security hardening approaches for Flink web are supported:

  - Whitelist filtering. Flink web can only be accessed through Yarn proxy.
  - Security header enhancement.
- In Flink clusters, ranges of listening ports of components can be configured.
- In HA mode, ACL control is supported.

12.6.4.2 Configuring Kafka

Sample project data of Flink is stored in Kafka. A user with Kafka permission can send data to Kafka and receive data from it.

Step 1 Ensure that clusters, including HDFS, Yarn, Flink, and Kafka are installed.

Step 2 Create a topic.

- Run Linux command line to create a topic. Before running commands, ensure that the kinit command, for example, **kinit flinkuser**, is run for authentication.

  NOTE

  To create a Flink user, you need to have the permission to create Kafka topics.

  The format of the command is shown as follows, in which **{zkQuorum}** indicates ZooKeeper cluster information and the format is *IP.port*, and **{Topic}** indicates the topic name.

  bin/kafka-topics.sh --create --zookeeper {zkQuorum}/kafka --replication-factor 1 --partitions 5 --topic {Topic}

  Assume the topic name is **topic 1**. The command for creating this topic is displayed as follows:

  ```
  /opt/client/Kafka/kafka/bin/kafka-topics.sh --create --zookeeper 10.96.101.32:2181,10.96.101.251:2181,10.96.101.177:2181,10.91.8.160:2181/kafka --replication-factor 1 --partitions 5 --topic topic1
  ```

- Configure the permission of the topic on the server.

  Set the **allow.everyone.if.no.acl.found** parameter of Kafka Broker to **true**.

Step 3 Perform the security authentication.

The Kerberos authentication, SSL encryption authentication, or Kerberos + SSL authentication mode can be used.

NOTE

For versions earlier than MRS 3.x, only Kerberos authentication is supported.

- **Kerberos authentication**
  - Client configuration

    In the Flink configuration file **flink-conf.yaml**, add configurations about Kerberos authentication. For example, add **KafkaClient** in **contexts** as follows:

    ```
    security.kerberos.login.keytab: /home/demo//keytab/flinkuser.keytab
    security.kerberos.login.principal: flinkuser
    security.kerberos.login.contexts: Client,KafkaClient
    security.kerberos.login.use-ticket-cache: false
    ```

    NOTE

    For versions earlier than MRS 3.x, set **security.kerberos.login.keytab** to **/home/demo/flink/release/keytab/flinkuser.keytab**.

  - Running parameter

    Running parameters about the **SASL_PLAINTEXT** protocol are as follows:

    ```
    --topic topic1 --bootstrap.servers 10.96.101.32:21007 --security.protocol SASL_PLAINTEXT  --sasl.kerberos.service.name kafka //10.96.101.32:21007 indicates the IP.port of the Kafka server.
    ```

- **SSL encryption**
  - Configure the server.

    Log in to FusionInsight Manager, choose **Cluster** > **Services** > **Kafka** > **Configurations**, and set **Type** to **All**. Search for **ssl.mode.enable** and set it to **true**.

  - Configure the client.

    i. Log in to FusionInsight Manager, choose **Cluster > *Name of the desired cluster* > Services > Kafka > More > Download Client** to download Kafka client.

    ii. Use the **ca.crt** certificate file in the client root directory to generate the **truststore** file for the client.

    Run the following command:

    ```
    keytool -noprompt -import -alias myservercert -file ca.crt -keystore truststore.jks
    ```

    The command execution result is similar to the following:

    ```
    drwx------.  5 zgd  users     4096 Feb  4 16:22 .
    drwxr-xr-x. 10 zgd  users     4096 Jan 22 17:38 ..
    -rwx------.  1 zgd  users      135 Jan 22 17:31 application.properties
    -rwx------.  1 zgd  users      790 Jan 22 17:31 bigdata_env.sample
    -rw-------.  1 zgd  users     1322 Jan 22 17:31 ca.crt
    -rwx------.  1 zgd  users     4508 Jan 22 17:31 conf.py
    -rw-------.  1 zgd  users      120 Jan 22 17:31 hosts
    -rwx------.  1 zgd  users      745 Jan 22 17:31 install.bat
    -rwx------.  1 zgd  users    15082 Jan 22 17:31 install.sh
    drwx------.  2 zgd  users     4096 Jan 22 17:38 JDK
    -rwx------.  1 zgd  users 37021723 Jan 22 17:31 jython-standalone-2.7.0.jar
    drwx------.  5 zgd  users     4096 Jan 22 17:38 Kafka
    drwx------.  3 zgd  users     4096 Jan 22 17:38 KrbClient
    -rwx------.  1 zgd  users      473 Jan 22 17:31 log4j.properties
    -rwx------.  1 zgd  users     2107 Jan 22 17:31 README
    -rwx------.  1 zgd  users     6949 Jan 22 17:31 refreshConfig.sh
    -rwx------.  1 zgd  users     1736 Jan 22 17:31 switchuser.py
    -rw-r--r--   1 root root      1004 Feb  4 16:22 truststore.jks
    ```

    iii. Run parameters.

    The value of **ssl.truststore.password** must be the same as the password you entered when creating **truststore**. Run the following command to run parameters:

    ```
    --topic topic1 --bootstrap.servers 10.96.101.32:9093 --security.protocol SSL --ssl.truststore.location /home/zgd/software/FusionInsight_Kafka_ClientConfig/truststore.jks --ssl.truststore.password XXX
    ```

- **Kerberos+SSL encryption**

  After completing preceding configurations of the client and server of Kerberos and SSL, modify the port number and protocol type in running parameters to enable the Kerberos+SSL encryption mode.

  ```
  --topic topic1 --bootstrap.servers 10.96.101.32:21009 --security.protocol SASL_SSL  --sasl.kerberos.service.name kafka --ssl.truststore.location /home/zgd/software/FusionInsight_Kafka_ClientConfig/truststore.jks --ssl.truststore.password XXX
  ```

----End

12.6.4.3 Configuring Pipeline

This section applies to MRS 3.x or later.

1. Configure files.
   - **nettyconnector.registerserver.topic.storage**: (Mandatory) Configures the path (on a third-party server) to information about IP address, port numbers, and concurrency of NettySink. For example:
     ```
     nettyconnector.registerserver.topic.storage: /flink/nettyconnector
     ```
   - **nettyconnector.sinkserver.port.range**: (Mandatory) Configures the range of port numbers of NettySink. For example:
     ```
     nettyconnector.sinkserver.port.range: 28444-28843
     ```
   - **nettyconnector.ssl.enabled**: Configures whether to enable SSL encryption between NettySink and NettySource. The default value is **false**. For example:
     ```
     nettyconnector.ssl.enabled: true
     ```
2. Configure security authentication.
   - SASL authentication of ZooKeeper depends on the HA configuration in the **flink-conf.yaml** file.
   - SSL configurations such as keystore, truststore, keystore password, truststore password, and password inherit from **flink-conf.yaml**. For details, see **Encrypted Transmission**.

12.6.5 Security Hardening

12.6.5.1 Authentication and Encryption

Security Authentication

Flink uses the following three authentication modes:

- Kerberos authentication: It is used between the Flink Yarn client and Yarn ResourceManager, JobManager and ZooKeeper, JobManager and HDFS, TaskManager and HDFS, Kafka and TaskManager, as well as TaskManager and ZooKeeper.
- Security cookie authentication: Security cookie authentication is used between Flink Yarn client and JobManager, JobManager and TaskManager, as well as TaskManager and TaskManager.
- Internal authentication of Yarn: The Internal authentication mechanism of Yarn is used between Yarn ResourceManager and ApplicationMaster (AM).

  NOTE

  - Flink JobManager and Yarn ApplicationMaster are in the same process.
  - If Kerberos authentication is enabled for the user's cluster, Kerberos authentication is required.
  - For versions earlier than MRS 3.*x*, Flink does not support security cookie authentication.

Table 12-82 Authentication modes

Authentication Mode	Description	Configuration Method

Kerberos authentication	Currently, only keytab authentication mode is supported.	1. Download the user keytab from the KDC server, and place the keytab to a directory on the host of the Flink client.
		2. Configure the following parameters in the **flink-conf.yaml** file:
		a. Keytab path
		`security.kerberos.login.keytab: /home/flinkuser/keytab/abc222.keytab`
		Note:
		/home/flinkuser/keytab/abc222.keytab indicates the user directory.
		b. Principal name
		`security.kerberos.login.principal: abc222`
		c. In HA mode, if ZooKeeper is configured, the Kerberos authentication configuration items must be configured as follows:
		`zookeeper.sasl.disable: false`
		`security.kerberos.login.contexts: Client`
		d. If you want to perform Kerberos authentication between Kafka client and Kafka broker, set the value as follows:
		`security.kerberos.login.contexts: Client,KafkaClient`

Authentication Mode	Description	Configuration Method
Security cookie authentication	-	1. In the **bin** directory of the Flink client, run the **generate_keystore.sh** script to generate security cookie, **flink.keystore**, and **flink.truststore**. Run the **sh generate_keystore.sh** command and enter the user-defined password. The password cannot contain **#**. **NOTE** After the script is executed, the **flink.keystore** and **flink.truststore** files are generated in the **conf** directory on the Flink client. In the **flink-conf.yaml** file, default values are specified for following parameters: ● Set **security.ssl.keystore** to the absolute path of the **flink.keystore** file. ● Set **security.ssl.truststore** to the absolute path of the **flink.truststore** file. ● Set **security.cookie** to a random password automatically generated by the **generate_keystore.sh** script. ● By default, **security.ssl.encrypt.enabled: false** is set in the **flink-conf.yaml** file by default. The **generate_keystore.sh** script sets **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password** to the password entered when the **generate_keystore.sh** script is called. ● For MRS 3.1.0 or later, if ciphertext is required and **security.ssl.encrypt.enabled** is set to **true** in the **flink-conf.yaml** file, the **generate_keystore.sh** script does not set **security.ssl.key-password**, **security.ssl.keystore-password**, and **security.ssl.truststore-password**. To obtain the values, use the Manager plaintext encryption API by running the following command: **curl -k -i -u** *Username*:*Password* **-X POST -HContent-type:application/json -d '{"plainText":"***Password***"}' 'https://***x.x.x.x***:28443/web/api/v2/tools/encrypt'** In the preceding command, *Username*:*Password* indicates the user name and password for logging in to the system. The password of **"plainText"** indicates the one used to call the **generate_keystore.sh** script. *x.x.x.x* indicates the floating IP address of Manager. 2. Set **security.enable: true** in the **flink-conf.yaml** file and check whether **security cookie** is configured successfully. Example: security.cookie: ae70acc9-9795-4c48-ad35-8b5adc8071744f605d1d-2726-432e-88ae-dd39bfec40a9

Authentication Mode	Description	Configuration Method
Internal authentication of Yarn	This authentication mode does not need to be configured by the user.	-

NOTE

One Flink cluster supports only one user. One user can create multiple Flink clusters.

Encrypted Transmission

Flink uses following encrypted transmission modes:

- Encrypted transmission inside Yarn: It is used between the Flink Yarn client and Yarn ResourceManager, as well as Yarn ResourceManager and JobManager.
- SSL transmission: SSL transmission is used between Flink Yarn client and JobManager, JobManager and TaskManager, as well as TaskManager and TaskManager.
- Encrypted transmission inside Hadoop: The internal encrypted transmission mode of Hadoop used between JobManager and HDFS, TaskManager and HDFS, JobManager and ZooKeeper, as well as TaskManager and ZooKeeper.

NOTE

Configuration about SSL encrypted transmission is mandatory while configuration about encryption of Yarn and Hadoop is not required.

To configure SSL encrypted transmission, configure the following parameters in the **flink-conf.yaml** file on the client:

1. Enable SSL and configure the SSL encryption algorithm. For MRS 3.x or later, see **Table 12-83**. Modify the parameters as required.

Table 12-83 Parameter description

Parameter	Example Value	Description
security.ssl.enabled	true	Enable SSL.

Parameter	Example Value	Description
akka.ssl.enabled	true	Enable Akka SSL.
blob.service.ssl.enabled	true	Enable SSL for the Blob channel.
taskmanager.data.ssl.enabled	true	Enable SSL transmissions between TaskManagers.
security.ssl.algorithms	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	Configure the SSL encryption algorithm.

For versions earlier than MRS 3.x, see **Table 12-84**.

Table 12-84 Parameter description

Parameter	Example Value	Description
security.ssl.internal.enabled	true	Enable internal SSL.
akka.ssl.enabled	true	Enable Akka SSL.
blob.service.ssl.enabled	true	Enable SSL for the Blob channel.
taskmanager.data.ssl.enabled	true	Enable SSL transmissions between TaskManagers.
security.ssl.algorithms	TLS_RSA_WITH_AES128 CBC_SHA256	Configure the SSL encryption algorithm.

For versions earlier than MRS 3.x, the following parameters in **Table 12-85** do not exist in the default Flink configuration of MRS. If you want to enable SSL for external connections, add the following parameters. After SSL for external connection is enabled, the native Flink page cannot be accessed using a Yarn proxy, because the Yarn open-source version cannot process HTTPS requests using a proxy. However, you can create a Windows VM in the same VPC of the cluster and access the native Flink page from the VM.

Table 12-85 Parameter description

Parameter	Example Value	Description
security.ssl.rest.enabled	true	Enable external SSL. If this parameter is set to **true**, set the related parameters by referring to **Table 12-85**.
security.ssl.rest.keystore	${path}/flink.keystore	Path for storing the **keystore**.
security.ssl.rest.keystore-password	-	A user-defined password of **keystore**.
security.ssl.rest.key-password	-	A user-defined password of the SSL key.
security.ssl.rest.truststore	${path}/flink.truststore	Path for storing the **truststore**.
security.ssl.rest.truststore-password	-	A user-defined password of **truststore**.

NOTE

Enabling SSL for data transmission between TaskManagers may pose great impact on the system performance.

2. In the **bin** directory of the Flink client, run the ***sh generate_keystore.sh*** *<password>* command. For details, see **Authentication and Encryption**. The configuration items in **Table 12-86** are set by default for MRS 3.*x* or later. You can also configure them manually.

Table 12-86 Parameter description

Parameter	Example Value	Description
security.ssl.keystore	${path}/flink.keystore	Path for storing the **keystore**. **flink.keystore** indicates the name of the **keystore** file generated by the **generate_keystore.sh*** tool.
security.ssl.keystore-password	-	A user-defined password of **keystore**.
security.ssl.key-password	-	A user-defined password of the SSL key.
security.ssl.truststore	${path}/flink.truststore	Path for storing the **truststore**. **flink.truststore** indicates the name of the **truststore** file generated by the **generate_keystore.sh*** tool.

Parameter	Example Value	Description
security.ssl.truststore-password	-	A user-defined password of **truststore**.

For versions earlier than MRS 3.x, the ***generate_keystore.sh*** command is generated automatically, and the configuration items in **Table 12-87** are set by default. You can also configure them manually.

Table 12-87 Parameter description

Parameter	Example Value	Description
security.ssl.internal.keystore	${path}/flink.keystore	Path for storing the **keystore**. **flink.keystore** indicates the name of the **keystore** file generated by the **generate_keystore.sh*** tool.
security.ssl.internal.keystore-password	-	A user-defined password of **keystore**.
security.ssl.internal.key-password	-	A user-defined password of the SSL key.
security.ssl.internal.truststore	${path}/flink.truststore	Path for storing the **truststore**. **flink.truststore** indicates the name of the **truststore** file generated by the **generate_keystore.sh*** tool.
security.ssl.internal.truststore-password	-	A user-defined password of **truststore**.

For versions earlier than MRS 3.x, if SSL for external connections is enabled, that is, **security.ssl.rest.enabled** is set to **true**, you need to configure the parameters listed in **Table 12-88**.

Table 12-88 Parameters

Parameter	Example Value	Description
security.ssl.rest.enabled	true	Enable external SSL. If this parameter is set to **true**, set the related parameters by referring to **Table 12-88**.
security.ssl.rest.keystore	${path}/flink.keystore	Path for storing the **keystore**.

Parameter	Example Value	Description
security.ssl.rest.keystore-password	-	A user-defined password of **keystore**.
security.ssl.rest.key-password	-	A user-defined password of the SSL key.
security.ssl.rest.truststore	${path}/flink.truststore	Path for storing the **truststore**.
security.ssl.rest.truststore-password	-	A user-defined password of **truststore**.

NOTE

The **path** directory is a user-defined directory for storing configuration files of the SSL keystore and truststore. The commands vary according to the relative path and absolute path. For details, see 3 and 4.

3. If the **keystore** or **truststore** file path is a relative path, the Flink client directory where the command is executed needs to access this relative path directly. Either of the following method can be used to transmit the keystore and truststore file:
   - Add **-t** option to the **CLI yarn-session.sh** command to transfer the **keystore** and **truststore** file to execution nodes. Example:
     ```
     ./bin/yarn-session.sh -t ssl/
     ```
   - Add **-yt** option to the **flink run** command to transfer the **keystore** and **truststore** file to execution nodes. Example:
     ```
     ./bin/flink run -yt ssl/ -ys 3  -m yarn-cluster -c
     org.apache.flink.examples.java.wordcount.WordCount /opt/client/Flink/flink/examples/batch/
     WordCount.jar
     ```

     NOTE

     - In the preceding example, **ssl/** is the sub-directory of the Flink client directory. It is used to store configuration files of the SSL keystore and truststore.
     - The relative path of **ssl/** must be accessible from the current path where the Flink client command is run.
4. If the keystore or truststore file path is an absolute path, the keystore and truststore files must exist in the absolute path on Flink Client and all nodes.

   NOTE

   For versions earlier than MRS 3.x, the user who submits the job must have the permission to read the keystore and truststore files.

   Either of the following methods can be used to execute applications. The **-t** or **-yt** option does not need to be added to transmit the **keystore** and **truststore** files.
   - Run the **CLI yarn-session.sh** command of Flink to execute applications. Example:
     ```
     ./bin/yarn-session.sh
     ```
   - Run the **Flink run** command to execute applications. Example:

```
./bin/flink run  -ys 3 -m yarn-cluster -c
org.apache.flink.examples.java.wordcount.WordCount /opt/client/Flink/flink/examples/batch/
WordCount.jar
```

12.6.5.2 ACL Control

In HA mode of Flink, ZooKeeper can be used to manage clusters and discover services. Zookeeper supports SASL ACL control. Only users who have passed the SASL (Kerberos) authentication have the permission to operate files on ZooKeeper. To enable SASL ACL control, perform following configurations in the Flink configuration file.

```
high-availability.zookeeper.client.acl: creator
zookeeper.sasl.disable: false
```

For details about configuration items, see **Table 12-78**.

12.6.5.3 Web Security

Coding Specifications

Note: The same coding mode is used on the web service client and server to prevent garbled characters and to enable input verification.

Security hardening: apply UTF-8 to response messages of web server.

Whitelist-based Filter of IP Addresses

Note: IP filter must be added to the web server to filter unauthorized requests from the source IP address and prevent unauthorized login.

Security: Add **jobmanager.web.allow-access-address** to enable the IP filter. By default, only Yarn users are supported.

NOTE

After the client is installed, you need to add the IP address of the client node to the **jobmanager.web.allow-access-address** configuration item.

Preventing Sending the Absolute Paths to the Client

Note: If an absolute path is sent to a client, the directory structure of the server is exposed, increasing the risk that attackers know and attack the system.

Security hardening: If the Flink configuration file contains a parameter starting with a slash (/), the first-level directory is deleted.

Same-origin Policy

The same-source policy applies to MRS 3.x or later.

If two URL protocols have same hosts and ports, they are of the same origin. Protocols of different origins cannot access each other, unless the source of the visitor is specified on the host of the service to be visited.

Security hardening: The default value of the header of the response header **Access-Control-Allow-Origin** is the IP address of ResourceManager on Yarn clusters. If the IP address is not from Yarn, mutual access is not allowed.

Defense Against XSS

Defense Against XSS applies to MRS 3.x or later.

Disabling XSS filter incurs the following risks:

- Leakage of sensitive information, for example, **sessionid**.
- Tampering of pages.
- Redirection to malicious websites.
- DoS attacks from other websites to Flink web.

Security hardening: Add the **X-XSS-Protection** security header to enable XSS filter. The default value configuration is **X-XSS-Protection: 1; mode=block**. If XSS attack is detected, rendering of the page is stopped. For example, if XSS attacks are detected in Internet Explorer 8, all contents of the page is replaced by **#**.

Preventing Sensitive Information Disclosure

Sensitive information disclosure prevention is applicable to MRS 3.x or later.

Web pages containing sensitive data must not be cached, to avoid leakage of sensitive information or data crosstalk among users who visit the internet through the proxy server.

Security hardening: Add **Cache-control**, **Pragma**, **Expires** security header. The default value is **Cache-Control: no-store**, **Pragma: no-cache**, and **Expires: 0**.

The security hardening stops contents interacted between Flink and web server from being cached.

Anti-Hijacking

Anti-hijacking applies to MRS 3.x or later.

Since hotlinking and clickjacking use framing technologies, security hardening is required to prevent attacks.

Security hardening: Add **X-Frame-Options** security header to specify whether the browser will load the pages from **iframe**, **frame** or **object**. The default value is **X-Frame-Options: DENY**, indicating that no pages can be nested to **iframe**, **frame** or **object**.

Logging calls of the Web Service APIs

This function applies to MRS 3.x or later.

Calls of the **Flink webmonitor restful** APIs are logged.

The **jobmanager.web.accesslog.enable** can be added in the **access log**. The default value is **true**. Logs are stored in a separate **webaccess.log** file.

Cross-Site Request Forgery Prevention

Cross-site request forgery (CSRF) prevention applies to MRS 3.x or later.

In **Browser/Server** applications, CSRF must be prevented for operations involving server data modification, such as adding, modifying, and deleting. The CSRF forces end users to execute non-intended operations on the current web application.

Security hardening: Only two post APIs, one delete API, and get interfaces are reserve for modification requests. All other APIs are deleted.

Troubleshooting

This function applies to MRS 3.x or later.

When the application is abnormal, exception information is filtered, logged, and returned to the client.

Security hardening

- A default error message page to filter information and log detailed error information.
- Four configuration parameters are added to ensure that the error page is switched to a specified URL provided by FusionInsight, preventing exposure of unnecessary information.

Table 12-89 Parameter description

Parameter	Description	Default Value	Mandatory
jobmanager.web.403-redirect-url	Web page access error 403. If 403 error occurs, the page switch to a specified page.	-	Yes
jobmanager.web.404-redirect-url	Web page access error 404. If 404 error occurs, the page switch to a specified page.	-	Yes
jobmanager.web.415-redirect-url	Web page access error 415. If 415 error occurs, the page switch to a specified page.	-	Yes
jobmanager.web.500-redirect-url	Web page access error 500. If 500 error occurs, the page switch to a specified page.	-	Yes

HTML5 Security

HTML5 security applies to MRS 3.x or later.

HTML5 is a next generation web development specification that provides new functions and extend the labels for developers. These new labels and functions increase the attack surface and pose attack risks (such as cross-domain resource sharing, client storage, WebWorker, WebRTC, and WebSocket).

Security hardening: Add the **Access-Control-Allow-Origin** parameter. For example, if you want to enable the cross-domain resource sharing, configure the **Access-Control-Allow-Origin** parameter of the HTTP response header.

NOTE

Flink does not involve security risks of functions such as storage on the client, WebWorker, WebRTC, and WebSocket.

12.6.6 Security Statement

- All security functions of Flink are provided by the open source community or self-developed. Security features that need to be configured by users, such as authentication and SSL encrypted transmission, may affect performance.
- As a big data computing and analysis platform, Flink does not detect sensitive information. Therefore, you need to ensure that the input data is not sensitive.
- You can evaluate whether configurations are secure as required.
- For any security-related problems, contact O&M support.

12.6.7 Using the Flink Web UI

12.6.7.1 Overview

12.6.7.1.1 Introduction to Flink Web UI

Flink web UI provides a web-based visual development platform. You only need to compile SQL statements to develop jobs, slashing the job development threshold. In addition, the exposure of platform capabilities allows service personnel to compile SQL statements for job development to quickly respond to requirements, greatly reducing the Flink job development workload.

NOTE

This section applies to only MRS 3.1.0 or later.

Flink Web UI Features

The Flink web UI has the following features:

- Enterprise-class visual O&M: GUI-based O&M management, job monitoring, and standardization of Flink SQL statements for job development.
- Quick cluster connection: After configuring the client and user credential key file, you can quickly access a cluster using the cluster connection function.

- Quick data connection: You can access a component by configuring the data connection function. If **Data Connection Type** is set to **HDFS**, you need to create a cluster connection. If **Authentication Mode** is set to **KERBEROS** for other data connection types, you need to create a cluster connection. If **Authentication Mode** is set to **SIMPLE**, you do not need to create a cluster connection.

  NOTE

  If **Data Connection Type** is set to **Kafka**, **Authentication Type** cannot be set to **KERBEROS**.
- Visual development platform: The input/output mapping table can be customized to meet the requirements of different input sources and output destinations.
- Easy to use GUI-based job management

Key Web UI Capabilities

Table 12-90 shows the key capabilities provided by Flink web UI.

Table 12-90 Key web UI capabilities

Item	Description
Batch-Stream convergence	• Batch jobs and stream jobs can be processed with a unified set of Flink SQL statements.
Flink SQL kernel capabilities	• Flink SQL supports customized window size, stream compute within 24 hours, and batch processing beyond 24 hours. • Flink SQL supports Kafka and HDFS reading. Data can be written to Kafka, Redis, and HDFS. The Redis dimension table can be joined. • A job can define multiple Flink SQL jobs, and multiple metrics can be combined into one job for computing. If a job contains same primary keys as well as same inputs and outputs, the job supports the computing of multiple windows. • The AVG, SUM, COUNT, MAX, and MIN statistical methods are supported.
Flink SQL functions on the console	• Cluster connection management allows you to configure clusters where services such as Kafka, Redis, and HDFS reside. • Data connection management allows you to configure services such as Kafka, Redis, and HDFS. • Data table management allows you to define data tables accessed by SQL statements and generate DDL statements. • Flink SQL job definition allows you to verify, parse, optimize, convert a job into a Flink job, and submit the job for running based on the entered SQL statements.

Item	Description
Flink job visual management	• Stream jobs and batch jobs can be defined in a visual manner. • Job resources, fault recovery policies, and checkpoint policies can be configured in a visual manner. • Status monitoring of stream and batch jobs are supported. • The Flink job O&M is enhanced, including redirection of the native monitoring page.
Performance and reliability	• Stream processing supports 24-hour window aggregation computing and millisecond-level performance. • Batch processing supports 90-day window aggregation computing, which can be completed in minutes. • Invalid data of stream processing and batch processing can be filtered out. • When HDFS data is read, the data can be filtered based on the calculation period in advance. • Data in Flink jobs comes from Redis. If fault recovery policies have been set for Flink jobs, data is read from Redis during calculation and no data is lost when a job is faulty. • If the job definition platform is faulty or the service is degraded, jobs cannot be redefined, but the computing of existing jobs is not affected. • The automatic restart mechanism is provided for job failures. You can configure restart policies.

12.6.7.1.2 Flink Web UI Application Process

The Flink web UI application process is shown as follows:

Figure 12-14 Application process

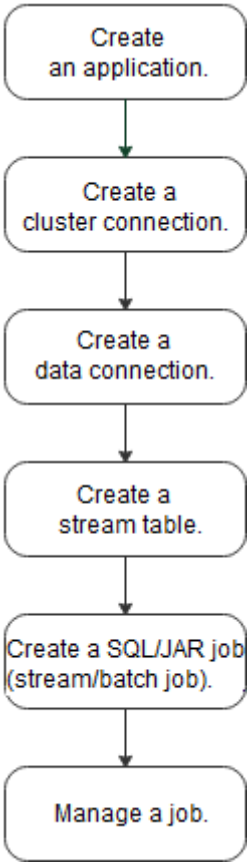


Table 12-91 Description of the Flink web UI application process

Phase	Description	Reference Section
Creating an application	Applications can be used to isolate different upper-layer services.	**Creating an Application on the Flink Web UI**
Creating a cluster connection	Different clusters can be accessed by configuring the cluster connection.	**Creating a Cluster Connection on the Flink Web UI**
Creating a data connection	Different data services can be accessed, such as HDFS, Kafka, and Redis, through the data connection.	**Creating a Data Connection on the Flink Web UI**
Creating a stream table	Data tables can be used to define basic attributes and parameters of source tables, dimension tables, and output tables.	**Managing Tables on the Flink Web UI**
Creating a SQL/JAR job (stream/batch job)	APIs can be used to define Flink jobs, including Flink SQL and Flink Jar jobs.	**Managing Jobs on the Flink Web UI**

Phase	Description	Reference Section
Managing a job	A created job can be managed, including starting, developing, stopping, deleting, and editing the job.	**Managing Jobs on the Flink Web UI**

12.6.7.2 FlinkServer Permissions Management

12.6.7.2.1 Overview

User **admin** of Manager does not have the FlinkServer service operation permission. To perform FlinkServer service operations, you need to grant related permission to the user.

Applications (tenants) in FlinkServer are the maximum management scope, including cluster connection management, data connection management, application management, stream table management, and job management.

There are three types of resource permissions for FlinkServer, as shown in **Table 12-92**.

Table 12-92 FlinkServer resource permissions

Name	Description	Remarks
Administrator permission	Users who have the permission can edit and view all applications.	This is the highest-level permission of FlinkServer. If you have the administrator permission, you have the permission on all applications by default.
Application edit permission	Users who have the permission can create, edit, and delete cluster connections and data connections. They can also create stream tables as well as create and run jobs.	In addition, users who have the permission can view current applications.
Application view permission	Users who have the permission can view applications.	-

12.6.7.2.2 Authentication Based on Users and Roles

This section describes how to create and configure a FlinkServer role on Manager as the system administrator. A FlinkServer role can be configured with administrator permission and the permissions to edit and view applications.

You need to set permissions for the specified user in FlinkServer so that they can update, query, and delete data.

Prerequisites

The administrator has planned permissions based on business needs.

Procedure

Step 1 Log in to Manager.

Step 2 Choose **System** > **Permission** > **Role**.

Step 3 On the displayed page, click **Create Role** and specify **Role Name** and **Description**.

Step 4 Set **Configure Resource Permission**.

FlinkServer permissions are as follows:

- **FlinkServer Admin Privilege**: highest-level permission. Users with the permission can perform service operations on all FlinkServer applications.
- **FlinkServer Application**: Users can set **application view** and **applications management** permissions on applications.

Table 12-93 Setting a role

Scenario	Role Authorization
Setting the administrator operation permission	In **Configure Resource Permission**, choose *Name of the desired cluster* > **Flink** and select **FlinkServer Admin Privilege**.
Setting a specified permission on applications	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Flink** > **FlinkServer Application**. 2. In the **Permission** column, select **application view** or **applications management**.

Step 5 Click **OK**. Return to role management page.

NOTE

After the FlinkServer is created, create a FlinkServer user and bind the user to the role and user group. For details, see .

----End

12.6.7.3 Accessing the Flink Web UI

Scenario

After Flink is installed in an MRS cluster, you can connect to clusters and data as well as manage stream tables and jobs using the Flink web UI.

This section describes how to access the Flink web UI in an MRS cluster.

NOTE

You are advised to use Google Chrome 50 or later to access the Flink web UI. The Internet Explorer may be incompatible with the Flink web UI.

Impact on the System

Site trust must be added to the browser when you access Manager and the Flink web UI for the first time. Otherwise, the Flink web UI cannot be accessed.

Procedure

Step 1 Log in to FusionInsight Manager as a user with **FlinkServer Admin Privilege**. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > **Services** > **Flink**.

Step 2 On the right of **Flink WebUI**, click the link to access the Flink web UI.

The Flink web UI provides the following functions:

- System management:
  - Cluster connection management allows you to create, view, edit, test, and delete a cluster connection.
  - Data connection management allows you to create, view, edit, test, and delete a data connection. Data connection types include HDFS, Kafka, and Redis.
  - Application management allows you to create, view, and delete an application.
- Stream table management allows you to create, view, edit, and delete a stream table.
- Job management allows you to create, view, start, develop, edit, stop, and delete a job.

----End

12.6.7.4 Creating an Application on the Flink Web UI

Scenario

Applications can be used to isolate different upper-layer services.

Creating an Application

Step 1 Access the Flink web UI as a user with **FlinkServer Admin Privilege**. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Application Management**.

Step 3 Click **Create Application**. On the displayed page, set parameters by referring to **Table 12-94** and click **OK**.

Table 12-94 Parameters for creating an application

Parameter	Description
Application	Name of the application to be created. The name can contain a maximum of 32 characters. Only letters, digits, and underscores (_) are allowed.
Description	Description of the application to be created. The value can contain a maximum of 85 characters.

After the application is created, you can switch to the application to be operated in the upper left corner of the Flink web UI and develop jobs.

----End

12.6.7.5 Creating a Cluster Connection on the Flink Web UI

Scenario

Different clusters can be accessed by configuring the cluster connection.

Creating a Cluster Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Cluster Connection Management**. The **Cluster Connection Management** page is displayed.

Step 3 Click **Create Cluster Connection**. On the displayed page, set parameters by referring to **Table 12-95** and click **OK**.

Table 12-95 Parameters for creating a cluster connection

Parameter	Description
Cluster Connection Name	Name of the cluster connection, which can contain a maximum of 100 characters. Only letters, digits, and underscores (_) are allowed.
Description	Description of the cluster connection name.
FusionInsight HD Version	Set a cluster version.
Secure Version	• If the secure version is used, select **Yes** for a security cluster. Enter the username and upload the user credential. • If not, select **No**.

Parameter	Description
Username	The user must have the minimum permissions for accessing services in the cluster. The name can contain a maximum of 100 characters. Only letters, digits, and underscores (_) are allowed. This parameter is available only when **Secure Version** is set to **Yes**.
Client Profile	Client profile of the cluster, in TAR format.
User Credential	User authentication credential in FusionInsight Manager in TAR format. This parameter is available only when **Secure Version** is set to **Yes**. Files can be uploaded only after the username is entered.

NOTE

To obtain the cluster client configuration files, perform the following steps:

1. Log in to FusionInsight Manager and choose **Cluster** > **Dashboard**.
2. Choose **More** > **Download Client** > **Configuration Files Only**, select a platform type, and click **OK**.

To obtain the user credential, perform the following steps:

1. Log in to FusionInsight Manager and click **System**.
2. In the **Operation** column of the user, choose **More** > **Download Authentication Credential**, select a cluster, and click **OK**.

----End

Editing a Cluster Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Cluster Connection Management**. The **Cluster Connection Management** page is displayed.

Step 3 In the **Operation** column of the item to be modified, click **Edit**. On the displayed page, modify the connection information by referring to **Table 12-95** and click **OK**.

----End

Testing a Cluster Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Cluster Connection Management**. The **Cluster Connection Management** page is displayed.

Step 3 In the **Operation** column of the item to be tested, click **Test**.

----End

Searching for a Cluster Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Cluster Connection Management**. The **Cluster Connection Management** page is displayed.

Step 3 In the upper right corner of the page, you can enter a search criterion to search for and view the cluster connection based on **Cluster Connection Name**.

----End

Deleting a Cluster Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Cluster Connection Management**. The **Cluster Connection Management** page is displayed.

Step 3 In the **Operation** column of the item to be deleted, click **Delete**, and click **OK** in the displayed page.

----End

12.6.7.6 Creating a Data Connection on the Flink Web UI

Scenario

Different data services can be accessed through data connections. Currently, FlinkServer supports HDFS, Kafka, and Redis data connections.

Creating a Data Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Data Connection Management**. The **Data Connection Management** page is displayed.

Step 3 Click **Create Data Connection**. On the displayed page, select a data connection type, enter information by referring to **Table 12-96**, and click **OK**.

Table 12-96 Parameters for creating a data connection

Parameter	Description	Example Value
Data Connection Type	Type of the data connection, which can be **HDFS**, **Kafka**, or **Redis**.	-
Data Connection Name	Name of the data connection, which can contain a maximum of 100 characters. Only letters, digits, and underscores (_) are allowed.	-

Parameter	Description	Example Value
Cluster Connection	Cluster connection name in configuration management. This parameter is mandatory for HDFS data connections and Redis data connections whose authentication type is **KERBEROS**.	-
Kafka broker	Connection information about Kafka broker instances. The format is *IP address*:*Port number*. Use commas (,) to separate multiple instances. This parameter is mandatory for Kafka data connections.	192.168.0.1:21005,192.168.0.2:21005
Redis Deployment Method	Redis deployment mode. Currently, only **Cluster** is supported. This parameter is mandatory for Redis data connections.	Cluster
Redis Server List	Connection information about Redis instances. The format is *IP address*:*Port number*. Use commas (,) to separate multiple instances. This parameter is mandatory for Redis data connections.	192.168.0.1:22400,192.168.0.2:22400
Authentication Mode	• **SIMPLE**: indicates that the connected service is in non-security mode and does not need to be authenticated. • **KERBEROS**: indicates that the connected service is in security mode and the Kerberos protocol for security authentication is used for authentication. This parameter is mandatory for Redis data connections.	-

----End

Editing a Data Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Data Connection Management**. The **Data Connection Management** page is displayed.

Step 3 In the **Operation** column of the item to be modified, click **Edit**. On the displayed page, modify the connection information by referring to **Table 12-96** and click **OK**.

----End

Testing a Data Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Data Connection Management**. The **Data Connection Management** page is displayed.

Step 3 In the **Operation** column of the item to be tested, click **Test**.

----End

Searching for a Data Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Data Connection Management**. The **Data Connection Management** page is displayed.

Step 3 In the upper right corner of the page, you can search for a data connection by name.

----End

Deleting a Data Connection

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Choose **System Management** > **Data Connection Management**. The **Data Connection Management** page is displayed.

Step 3 In the **Operation** column of the item to be deleted, click **Delete**, and click **OK** in the displayed page.

----End

12.6.7.7 Managing Tables on the Flink Web UI

Scenario

Data tables can be used to define basic attributes and parameters of source tables, dimension tables, and output tables.

Creating a Stream Table

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Table Management**. The table management page is displayed.

Step 3 Click **Create Stream Table**. On the stream table creation page, set parameters by referring to **Table 12-97** and click **OK**.

Table 12-97 Parameters for creating a stream table

Parameter	Description	Remarks
Stream/Table Name	Stream/Table name, which can contain 1 to 64 characters. Only letters, digits, and underscores (_) are allowed.	Example: **flink_sink**
Description	Stream/Table description information, which can contain 1 to 1024 characters.	-
Mapping Table Type	Flink SQL does not provide the data storage function. Table creation is actually the creation of mapping for external data tables or storage. The value can be **Kafka**, **HDFS**, or **Redis**.	-
Type	Includes the data source table **Source**, data result table **Sink**, and data dimension table **Table**. Tables included in different mapping table types are as follows: ● Kafka: **Source** and **Sink** ● HDFS: **Source** and **Sink** ● Redis: **Sink** and **Table**	-
Data Connection	Name of the data connection.	-
Topic	Kafka topic to be read. Multiple Kafka topics can be read. Use separators to separate topics. This parameter is available when **Mapping Table Type** is set to **Kafka**.	-
File Path	HDFS directory or a single file path to be transferred. This parameter is available when **Mapping Table Type** is set to **HDFS**.	Example: **/user/sqoop/** or **/user/sqoop/example.csv**
Code	Codes corresponding to different mapping table types are as follows: ● Kafka: **CSV** and **JSON** ● HDFS: **CSV** ● Redis: – If **Type** is set to **Sink**, the value can be **String**, **List**, **Set**, **Zset**, or **Hash**. – If **Type** is set to **Table**, the value can be **String** or **Zset**.	-

Parameter	Description	Remarks
Prefix	When **Mapping Table Type** is set to **Kafka**, **Type** is set to **Source**, and **Code** is set to **JSON**, this parameter indicates the hierarchical prefixes of multi-layer nested JSON, which are separated by commas (,).	For example, **data,info** indicates that the content under **data** and **info** in the nested JSON file is used as the data input in JSON format.
	If **Mapping Table Type** is set to **Redis**, prefixes will be automatically added to the key or you can manually enter prefixes.	For example, if the key value is **key1** and the prefix is **test**, the key written to Redis is **test:key1**.
Separator	Meanings of this parameter corresponding to different mapping table types are as follows: ● Kafka: This parameter is used to specify the separator between CSV fields. This parameter is available when **Code** is set to **CSV**. ● Redis: This parameter is used to specify the field separator.	Example: comma (**,**)
Row Separator	Line break in the file, including **\r**, **\n**, and **\r\n**. This parameter is available when **Mapping Table Type** is set to **HDFS**.	-
Column Separator	Field separator in the file. This parameter is available when **Mapping Table Type** is set to **HDFS**.	Example: comma (**,**)
Data Validity Period	Data validity period, which can be **Permanent**, **Effective Duration**, or **Deadline**. This parameter is available when **Mapping Table Type** is set to **Redis** and **Type** is set to **Sink**.	-
Stream Table Structure	Stream/Table structure, including **Name** and **Type**.	-
Proctime	System time, which is irrelevant to the data timestamp. That is, the time when the calculation is complete in Flink operators. This parameter is available when **Type** is set to **Source**.	-

Parameter	Description	Remarks
Event Time	Time when an event is generated, that is, the timestamp generated during data generation. This parameter is available when **Type** is set to **Source**.	-

----End

Editing a Stream Table

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Table Management**. The table management page is displayed.

Step 3 In the **Operation** column of the item to be modified, click **Edit**. On the displayed page, modify the stream table information by referring to **Table 12-97** and click **OK**.

----End

Searching for a stream table

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Table Management**. The table management page is displayed.

Step 3 In the upper right corner of the page, you can enter a keyword to search for stream table information.

----End

Deleting a Stream Table

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Table Management**. The table management page is displayed.

Step 3 In the **Operation** column of the item to be deleted, click **Delete**, and click **OK** in the displayed page.

----End

12.6.7.8 Managing Jobs on the Flink Web UI

Scenario

Define Flink jobs, including Flink SQL and Flink JAR jobs.

Creating a Stream Table

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 Click **Create Job**. On the displayed job creation page, set parameters by referring to **Table 12-98** and click **OK**. The job development page is displayed.

Table 12-98 Parameters for creating a job

Parameter	Description
Type	Job type, which can be **Flink SQL** or **Flink Jar**.
Name	Job name, which can contain a maximum of 64 characters. Only letters, digits, and underscores (_) are allowed.
Task Type	Type of the job data source, which can be a stream job or a batch job.
Description	Job description, which can contain a maximum of 100 characters.

Step 4 (Optional) If you need to develop a job immediately, configure the job on the job development page.

- Creating a Flink SQL job
  a. Develop the job on the job development page.
  b. Click **Check Semantic** to check the input content and click **Format SQL** to format SQL statements.
  c. After the job SQL statements are developed, set basic and customized parameters as required by referring to **Table 12-99** and click **Save**.

Table 12-99 Basic parameters

Parameter	Description
Parallelism	Number of concurrent jobs. The value must be a positive integer containing a maximum of 64 characters.
Maximum Operator Parallelism	Maximum parallelism of operators. The value must be a positive integer containing a maximum of 64 characters.
JobManager Memory (MB)	Memory of JobManager The minimum value is **512** and the value can contain a maximum of 64 characters.
Submit Queue	Queue to which a job is submitted. If this parameter is not set, the **default** queue is used. The queue name can contain a maximum of 30 characters. Only letters, digits, and underscores (_) are allowed.

Parameter	Description
taskManager	taskManager running parameters include: ■ **Slots**: If this parameter is left blank, the default value **1** is used. ■ **Memory (MB)**: The minimum value is **512**.
Enable CheckPoint	Whether to enable CheckPoint. After CheckPoint is enabled, you need to configure the following information: ■ **Time Interval (ms)**: This parameter is mandatory. ■ **Mode**: This parameter is mandatory. The options are **EXACTLY_ONCE** and **AT_LEAST_ONCE**. ■ **Minimum Interval (ms)**: The minimum value is **10**. ■ **Timeout Duration**: The minimum value is **10**. ■ **Maximum Parallelism**: The value must be a positive integer containing a maximum of 64 characters. ■ **Whether to clean up**: This parameter can be set to **Yes** or **No**. ■ **Whether to enable incremental checkpoints**: This parameter can be set to **Yes** or **No**.
Failure Recovery Policy	Failure recovery policy of a job. The options are as follows: ■ **fixed-delay**: You need to configure **Retry Times** and **Retry Interval (s)**. ■ **failure-rate**: You need to configure **Max Retry Times**, **Interval (min)**, and **Retry Interval (s)**. ■ **none**

d. Click **Submit** in the upper left corner to submit the job.

- Creating a Flink JAR job
  a. Click **Select** to upload a local JAR file and set parameters by referring to **Table 12-100** or add customized parameters.

Table 12-100 Parameter configuration

Parameter	Description
Local .jar File	Upload a local JAR file. The size of the file cannot exceed 10 MB.
Main Class	Main-Class type. ■ **Default**: By default, the class name is specified based on the **Mainfest** file in the JAR file. ■ **Specify**: Manually specify the class name.
Type	Class name. This parameter is available when **Main Class** is set to **Specify**.
Class Parameter	Class parameters of Main-Class (parameters are separated by spaces).
Parallelism	Number of concurrent jobs. The value must be a positive integer containing a maximum of 64 characters.
JobManager Memory (MB)	Memory of JobManager The minimum value is **512** and the value can contain a maximum of 64 characters.
Submit Queue	Queue to which a job is submitted. If this parameter is not set, the **default** queue is used. The queue name can contain a maximum of 30 characters. Only letters, digits, and underscores (_) are allowed.
taskManager	taskManager running parameters include: ■ **Slots**: If this parameter is left blank, the default value **1** is used. ■ **Memory (MB)**: The minimum value is **512**.

b. Click **Save** to save the configuration and click **Submit** to submit the job.

Step 5 Return to the job management page. You can view information about the created job, including job name, type, status, kind, and description.

----End

Starting a Job

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the **Operation** column of the job to be started, click **Start** to run the job. Jobs in the **Draft**, **Saved**, **Submission failed**, **Running succeeded**, **Running failed**, or **Stop** state can be started.

----End

Developing a Job

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the **Operation** column of the job to be developed, click **Develop** to go to the job development page. Develop a job by referring to **Step 4**. You can view created stream tables and fields in the list on the left.

----End

Editing the Job Name and Description

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the **Operation** column of the item to be modified, click **Edit**, modify **Description**, and click **OK** to save the modification.

----End

Viewing Job Details

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the **Operation** column of the item to be viewed, choose **More** > **Job Monitoring** to view the job running details.

NOTE

You can only view details about jobs in the **Running** state.

----End

Checkpoint Failure Recovery

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the Operation column of the item to be restored, click **More** > **Checkpoint Failure Recovery**. You can perform checkpoint failure recovery for jobs in the **Running failed**, **Running Succeeded**, or **Stop** state.

----End

Filtering/Searching for Jobs

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the upper right corner of the page, you can obtain job information by selecting the job name, or enter a keyword to search for a job.

----End

Stopping a Job

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the **Operation** column of the item to be stopped, click **Stop**. Jobs in the **Submitting**, **Submission succeeded**, or **Running** state can be stopped.

----End

Deleting a Job

Step 1 Access the Flink web UI. For details, see **Accessing the Flink Web UI**.

Step 2 Click **Job Management**. The job management page is displayed.

Step 3 In the **Operation** column of the item to be deleted, click **Delete**, and click **OK** in the displayed page. Jobs in the **Draft**, **Saved**, **Submission failed**, **Running succeeded**, **Running failed**, or **Stop** state can be deleted.

----End

12.6.8 Flink Log Overview

Log Description

Log path:

- Run logs of a Flink job: **${BIGDATA_DATA_HOME}/hadoop/data${i}/nm/containerlogs/application_${appid}/container_{$contid}**

  NOTE

  The logs of executing tasks are stored in the preceding path. After the execution is complete, the Yarn configuration determines whether these logs are gathered to the HDFS directory.

- FlinkResource run logs: **/var/log/Bigdata/flink/flinkResource**

Log archive rules:

1. FlinkResource run logs:
   - By default, service logs are backed up each time when the log size reaches 20 MB. A maximum of 20 logs can be reserved without being compressed.

NOTE

For versions earlier than MRS 3.x, The executor logs are backed up each time when the log size reaches 30 MB. A maximum of 20 logs can be reserved without being compressed.

- You can set the log size and number of compressed logs on the Manager page or modify the corresponding configuration items in **log4j-cli.properties**, **log4j.properties**, and **log4j-session.properties** in **/opt/client/Flink/flink/conf/** on the client. **/opt/client** is the client installation directory.

Table 12-101 FlinkResource log list

Type	Name	Description
FlinkResource run logs	checkService.log	Health check log
	kinit.log	Initialization log
	postinstall.log	Service installation log
	prestart.log	Prestart script log
	start.log	Startup log

Log Level

Table 12-102 describes the log levels supported by Flink. The priorities of log levels are ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-102 Log levels

Level	Description
ERROR	Error information about the current event processing
WARN	Exception information about the current event processing
INFO	Normal running status information about the system and events
DEBUG	System information and system debugging information

To modify log levels, perform the following steps:

Step 1 Go to the **All Configurations** page of Flink by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

----End

NOTE

- After the configuration is complete, you do not need to restart the service. Download the client again for the configuration to take effect.
- You can also change the configuration items corresponding to the log level in **log4j-cli.properties**, **log4j.properties**, and **log4j-session.properties** in **/opt/client/Flink/flink/conf/** on the client. **/opt/client** is the client installation directory.
- When a job is submitted using a client, a log file is generated in the **log** folder on the client. The default umask value is **0022**. Therefore, the default log permission is **644**. To change the file permission, you need to change the umask value. For example, to change the umask value of user **omm**:
  - Add **umask 0026** to the end of the **/home/omm/.baskrc** file.
  - Run the **source /home/omm/.baskrc** command to make the file permission take effect.

Log Format

Table 12-103 Log formats

Type	Format	Example
Run log	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2019-06-27 21:30:31,778 | INFO | [flink-akka.actor.default-dispatcher-3] | TaskManager container_e10_1498290698388_0004_02_000007 has started. | org.apache.flink.yarn.YarnFlinkResourceManager (FlinkResourceManager.java:368)

12.6.9 Flink Performance Tuning

12.6.9.1 Optimization DataStream

12.6.9.1.1 Memory Configuration Optimization

Scenarios

The computing of Flink depends on memory. If the memory is insufficient, the performance of Flink will be greatly deteriorated. One solution is to monitor garbage collection (GC) to evaluate the memory usage. If the memory becomes the performance bottleneck, optimize the memory usage according to the actual situation.

If **Full GC** is frequently reported in the Container GC on the Yarn that monitors the node processes, the GC needs to be optimized.

NOTE

In the **env.java.opts** configuration item of the **conf/flink-conf.yaml** file on the client, add the **-Xloggc:<LOG_DIR>/gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=20M** parameter. The GC log is configured by default.

Procedure

- Optimize GC.

  Adjust the ratio of tenured generation memory to young generation memory. In the **conf/flink-conf.yaml** configuration file on the client, add the **-XX:NewRatio** parameter to the **env.java.opts** configuration item. For example, **-XX:NewRatio=2** indicates that ratio of tenured generation memory to young generation memory is 2:1, that is, the young generation memory occupies one third and tenured generation memory occupies two thirds.
- When developing Flink applications, optimize the partitioning or grouping operation of DataStream.
  - If partitioning causes data skew, partitions need to be optimized.
  - Do not perform concurrent operations, because some operations, WindowAll for example, to DataStream do not support parallelism.
  - Do not use set keyBy to string type.

12.6.9.1.2 Configuring DOP

Scenario

The degree of parallelism (DOP) indicates the number of tasks to be executed concurrently. It determines the number of data blocks after the operation. Configuring the DOP will optimize the number of tasks, data volume of each task, and the host processing capability.

Query the CPU and memory usage. If data and tasks are not evenly distributed among nodes, increase the DOP for even distribution.

Procedure

Configure the DOP at one of the following layers (the priorities of which are in the descending order) based on the actual memory, CPU, data, and application logic conditions:

- Operator

  Call the **setParallelism()** method to specify the DOP of an operator, data source, and sink. For example:

```
final StreamExecutionEnvironment env = StreamExecutionEnvironment.getExecutionEnvironment();

DataStream<String> text = [...]
DataStream<Tuple2<String, Integer>> wordCounts = text
    .flatMap(new LineSplitter())
    .keyBy(0)
    .timeWindow(Time.seconds(5))
```

```
    .sum(1).setParallelism(5);

wordCounts.print();

env.execute("Word Count Example");
```

- Execution environment

  Flink runs in the execution environment which defines a default DOP for operators, data source and data sink.

  Call the **setParallelism()** method to specify the default DOP of the execution environment. Example:

```
final StreamExecutionEnvironment env = StreamExecutionEnvironment.getExecutionEnvironment();
env.setParallelism(3);
DataStream<String> text = [...]
DataStream<Tuple2<String, Integer>> wordCounts = [...]
wordCounts.print();
env.execute("Word Count Example");
```

- Client

  Specify the DOP when submitting jobs to Flink on the client. If you use the CLI client, specify the DOP using the **-p** parameter. Example:

```
./bin/flink run -p 10 ../examples/*WordCount-java*.jar
```

- System

  On the Flink client, modify the **parallelism.default** parameter in the **flink-conf.yaml** file under the conf to specify the DOP for all execution environments.

12.6.9.1.3 Configuring Process Parameters

Scenario

In Flink on Yarn mode, there are JobManagers and TaskManagers. JobManagers and TaskManagers schedule and run tasks.

Therefore, configuring parameters of JobManagers and TaskManagers can optimize the execution performance of a Flink application. Perform the following steps to optimize the Flink cluster performance.

Procedure

Step 1 Configure JobManager memory.

JobManagers are responsible for task scheduling and message communications between TaskManagers and ResourceManagers. JobManager memory needs to be increased as the number of tasks and the DOP increases.

JobManager memory needs to be configured based on the number of tasks.

- When running the **yarn-session** command, add the **-jm MEM** parameter to configure the memory.
- When running the **yarn-cluster** command, add the **-yjm MEM** parameter to configure the memory.

Step 2 Configure the number of TaskManagers.

Each core of a TaskManager can run a task at the same time. Increasing the number of TaskManagers has the same effect as increasing the DOP. Therefore,

you can increase the number of TaskManagers to improve efficiency when there are sufficient resources.

Step 3 Configure the number of TaskManager slots.

Multiple cores of a TaskManager can process multiple tasks at the same time. This has the same effect as increasing the DOP. However, the balance between the number of cores and the memory must be maintained, because all cores of a TaskManager share the memory.

- When running the **yarn-session** command, add the **-s NUM** parameter to configure the number of slots.
- When running the **yarn-cluster** command, add the **-ys NUM** parameter to configure the number of slots.

Step 4 Configure TaskManager memory.

TaskManager memory is used for task execution and communication. A large-size task requires more resources. In this case, you can increase the memory.

- When running the **yarn-session** command, add the **-tm MEM** parameter to configure the memory.
- When running the **yarn-cluster** command, add the **-ytm MEM** parameter to configure the memory.

----End

12.6.9.1.4 Optimizing the Design of Partitioning Method

Scenarios

The divide of tasks can be optimized by optimizing the partitioning method. If data skew occurs in a certain task, the whole execution process is delayed. Therefore, when designing the partitioning method, ensure that partitions are evenly assigned.

Procedure

Partitioning methods are as follows:

- **Random partitioning**: randomly partitions data.
  ```
  dataStream.shuffle();
  ```
- **Rebalancing (round-robin partitioning)**: evenly partitions data based on round-robin. The partitioning method is useful to optimize data with data skew.
  ```
  dataStream.rebalance();
  ```
- **Rescaling**: assign data to downstream subsets in the form of round-robin. The partitioning method is useful if you want to deliver data from each parallel instance of a data source to subsets of some mappers without the using rebalance (), that is, the complete rebalance operation.
  ```
  dataStream.rescale();
  ```
- **Broadcast**: broadcast data to all partitions.
  ```
  dataStream.broadcast();
  ```

- **User-defined partitioning**: use a user-defined partitioner to select a target task for each element. The user-defined partitioning allows user to partition data based on a certain feature to achieve optimized task execution.

  The following is an example:

```
// fromElements builds simple Tuple2 stream
DataStream<Tuple2<String, Integer>> dataStream = env.fromElements(Tuple2.of("hello",1),
Tuple2.of("test",2), Tuple2.of("world",100));

// Defines the key value used for partitioning. Adding one to the value equals to the id.
Partitioner<Tuple2<String, Integer>> strPartitioner = new Partitioner<Tuple2<String, Integer>>() {
    @Override
    public int partition(Tuple2<String, Integer> key, int numPartitions) {
        return (key.f0.length() + key.f1) % numPartitions;
    }
};

// The Tuple2 data is used as the basis for partitioning.

dataStream.partitionCustom(strPartitioner, new KeySelector<Tuple2<String, Integer>, Tuple2<String,
Integer>>() {
    @Override
    public Tuple2<String, Integer> getKey(Tuple2<String, Integer> value) throws Exception {
        return value;
    }
}).print();
```

12.6.9.1.5 Configuring the Netty Network Communication

Scenarios

The communication of Flink is based on Netty network. The network performance determines the data switching speed and task execution efficiency. Therefore, the performance of Flink can be optimized by optimizing the Netty network.

Procedure

In the **conf/flink-conf.yaml** file on the client, change configurations as required. Exercise caution when changing default values, because default values are optimal.

- **taskmanager.network.netty.num-arenas**: Specifies the number of arenas of Netty. The default value is **taskmanager.numberOfTaskSlots**.
- **taskmanager.network.netty.server.numThreads** and **taskmanager.network.netty.client.numThreads**: specify the number of threads on the client and server. The default value is **taskmanager.numberOfTaskSlots**.
- **taskmanager.network.netty.client.connectTimeoutSec**: specifies the timeout interval for connection of TaskManager client. The default value is **120s**.
- **taskmanager.network.netty.sendReceiveBufferSize**: specifies the buffer size of the Netty network. The default value is the buffer size (cat /proc/sys/net/ipv4/tcp_[rw]mem) of the system and the value is usually 4 MB.
- **taskmanager.network.netty.transport**: specifies the transmission method of the Netty network. The default value is **nio**. The value can only be **nio** and **epoll**.

12.6.9.1.6 Experience Summary

Avoiding Data Skew

If data skew occurs (certain data volume is extremely large), the execution time of tasks is inconsistent even though no GC is performed.

- Redefine keys. Use keys of smaller granularity to optimize the task size.
- Modify the DOP.
- Call the rebalance operation to balance data partitions.

Setting Timeout Interval for the Buffer

- During the execution of tasks, data is exchanged through network. You can set the **setBufferTimeout** parameter to specify a buffer timeout interval for data exchanging among different servers.
- If **setBufferTimeout** is set to **-1**, the refreshing operation is performed when the buffer is full to maximize the throughput. If **setBufferTimeout** is set to **0**, the refreshing operation is performed each time data is received to minimize the delay. If **setBufferTimeout** is set to a value greater than **0**, the refreshing operation is performed after the buffer times out.

  The following is an example:

```
env.setBufferTimeout(timeoutMillis);

env.generateSequence(1,10).map(new MyMapper()).setBufferTimeout(timeoutMillis);
```

12.6.10 Common Flink Shell Commands

This section applies to MRS 3.x or later.

Before running the Flink shell commands, perform the following steps:

Step 1 Install the Flink client in a directory, for example, **/opt/client**.

Step 2 Run the following command to initialize environment variables:

source /opt/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled, skip this step.

kinit *Service user*

Step 4 Run the related commands according to **Table 12-104**.

Table 12-104 Flink Shell commands

Command	Description	Description
yarn-session.sh	**-at,--applicationType <arg>**: Defines the Yarn application type. **-D <property=value>**: Configures dynamic parameter. **-d,--detached**: Disables the interactive mode and starts a separate Flink Yarn session. **-h,--help**: Displays the help information about the Yarn session CLI. **-id,--applicationId <arg>**: Binds to a running Yarn session. **-j,--jar <arg>**: Sets the path of the user's JAR file. **-jm,--jobManagerMemory <arg>**: Sets the JobManager memory. **-m,--jobmanager <arg>**: Address of the JobManager (master) to which to connect. Use this parameter to connect to a specified JobManager. **-nl,--nodeLabel <arg>**: Specifies the nodeLabel of the Yarn application. **-nm,--name <arg>**: Customizes a name for the application on Yarn. **-q,--query**: Queries available Yarn resources. **-qu,--queue <arg>**: Specifies a Yarn queue. **-s,--slots <arg>**: Sets the number of slots for each TaskManager. **-t,--ship <arg>**: specifies the directory of the file to be sent. **-tm,--taskManagerMemory <arg>**: sets the TaskManager memory. **-yd,--yarndetached**: starts Yarn in the detached mode. **-z,--zookeeperNamespace <args>**: specifies the namespace of ZooKeeper. **-h**: Gets help information.	Start a resident Flink cluster to receive tasks from the Flink client.

Command	Description	Description
flink run	**-c,--class <classname>**: Specifies a class as the entry for running programs. **-C,--classpath <url>**: Specifies **classpath**. **-d,--detached**: Runs a job in the detached mode. **-files,--dependencyFiles <arg>**: File on which the Flink program depends. **-n,--allowNonRestoredState**: A state that cannot be restored can be skipped during restoration from a snapshot point in time. For example, if an operator in the program is deleted, you need to add this parameter when restoring the snapshot point. **-m,--jobmanager <host:port>**: Specifies the JobManager. **-p,--parallelism <parallelism>**: Specifies the job DOP, which will overwrite the DOP parameter in the configuration file. **-q,--sysoutLogging**: Disables the function of outputting Flink logs to the console. **-s,--fromSavepoint <savepointPath>**: Specifies a savepoint path for recovering jobs. **-z,--zookeeperNamespace <zookeeperNamespace>**: specifies the namespace of ZooKeeper. **-yat,--yarnapplicationType <arg>**: Defines the Yarn application type. **-yD <arg>**: Dynamic parameter configuration. **-yd,--yarndetached**: Starts Yarn in the detached mode. **-yh,--yarnhelp**: Obtains the Yarn help. **-yid,--yarnapplicationId <arg>**: Binds a job to a Yarn session. **-yj,--yarnjar <arg>**: Sets the path to Flink jar file. **-yjm,--yarnjobManagerMemory <arg>**: Sets the JobManager memory (MB). **-ynm,--yarnname <arg>**: Customizes a name for the application on Yarn. **-yq,--yarnquery**: Queries available Yarn resources (memory and CPUs).	Submit a Flink job. 1. The **-y*** parameter is used in the **yarn-cluster** mode. 2. If the parameter is not **-y***, you need to run the **yarn-session** command to start the Flink cluster before running this command to submit a task.

Command	Description	Description
	-yqu,--yarnqueue <arg>: Specifies a Yarn queue. **-ys,--yarnslots**: Sets the number of slots for each TaskManager. **-yt,--yarnship <arg>**: Specifies the path of the file to be sent. **-ytm,--yarntaskManagerMemory <arg>**: Sets the TaskManager memory (MB). **-yz,--yarnzookeeperNamespace <arg>**: Specifies the namespace of ZooKeeper. The value must be the same as the value of **yarn-session.sh -z**. **-h**: Gets help information.	
flink info	**-c,--class <classname>**: Specifies a class as the entry for running programs. **-p,--parallelism <parallelism>**: Specifies the DOP for running programs. **-h**: Gets help information.	Display the execution plan (JSON) of the running program.
flink list	**-a,--all**: displays all jobs. **-m,--jobmanager <host:port>**: specifies the JobManager. **-r,--running:** displays only jobs in the running state. **-s,--scheduled**: displays only jobs in the scheduled state. **-z,--zookeeperNamespace <zookeeperNamespace>**: specifies the namespace of ZooKeeper. **-yid,--yarnapplicationId <arg>**: binds a job to a Yarn session. **-h**: gets help information.	Query running programs in the cluster.

Command	Description	Description
flink stop	**-d,--drain**: sends MAX_WATERMARK before the savepoint is triggered and the job is stopped. **-p,--savepointPath <savepointPath>**: path for storing savepoints. The default value is **state.savepoints.dir**. **-m,--jobmanager <host:port>**: specifies the JobManager. **-z,--zookeeperNamespace <zookeeperNamespace>**: specifies the namespace of ZooKeeper. **-yid,--yarnapplicationId <arg>**: binds a job to a Yarn session. **-h**: gets help information.	Forcibly stop a running job (only streaming jobs are supported. **StoppableFunction** needs to be implemented on the source side in service code).
flink cancel	**-m,--jobmanager <host:port>**: specifies the JobManager. **-s,--withSavepoint <targetDirectory>**: triggers a savepoint when a job is canceled. The default directory is **state.savepoints.dir**. **-z,--zookeeperNamespace <zookeeperNamespace>**: specifies the namespace of ZooKeeper. **-yid,--yarnapplicationId <arg>**: binds a job to a Yarn session. **-h**: gets help information.	Cancel a running job.
flink savepoint	**-d,--dispose <arg>**: specifies a directory for storing the savepoint. **-m,--jobmanager <host:port>**: specifies the JobManager. **-z,--zookeeperNamespace <zookeeperNamespace>**: specifies the namespace of ZooKeeper. **-yid,--yarnapplicationId <arg>**: binds a job to a Yarn session. **-h**: gets help information.	Trigger a savepoint.

Command	Description	Description
source *Client installation directory*/**bigdata_env**	None	Import client environment variables. Restriction: If the user uses a custom script (for example, **A.sh**) and runs this command in the script, variables cannot be imported to the **A.sh** script. If variables need to be imported to the custom script **A.sh**, the user needs to use the secondary calling method. For example, first call the **B.sh** script in the **A.sh** script, and then run this command in the **B.sh** script. Parameters can be imported to the **A.sh** script but cannot be imported to the **B.sh** script.
start-scala-shell.sh	local | remote <host> <port> | yarn: running mode	Start the scala shell.
sh generate_keystore.sh	-	Run the **generate_keystore.sh** script to generate security cookie, **flink.keystore**, and **flink.truststore**. You need to enter a user-defined password that does not contain number signs (#).

----End

12.6.11 Reference

12.6.11.1 Example of Issuing a Certificate

Generate the **generate_keystore.sh** script based on the sample code and save the script to the **bin** directory on the Flink client.

```
#!/bin/bash

KEYTOOL=${JAVA_HOME}/bin/keytool
KEYSTOREPATH="$FLINK_HOME/conf/"
CA_ALIAS="ca"
CA_KEYSTORE_NAME="ca.keystore"
CA_DNAME="CN=Flink_CA"
CA_KEYALG="RSA"
CLIENT_CONF_YAML="$FLINK_HOME/conf/flink-conf.yaml"
KEYTABPRINCEPAL=""

function getConf()
{
    if [ $# -ne 2 ]; then
        echo "invalid parameters for getConf"
        exit 1
    fi

    confName="$1"
    if [ -z "$confName" ]; then
        echo "conf name is empty."
        exit 2
    fi

    configFile=$FLINK_HOME/conf/client.properties
    if [ ! -f $configFile ]; then
        echo $configFile" is not exist."
        exit 3
    fi

    defaultValue="$2"
    cnt=$(grep $1 $configFile | wc -l)
    if [ $cnt -gt 1 ]; then
        echo $confName" has multi values in "$configFile
        exit 4
    elif [ $cnt -lt 1 ]; then
        echo $defaultValue
    else
        line=$(grep $1 $configFile)
        confValue=$(echo "${line#*=}")
        echo "$confValue"
    fi
}

function createSelfSignedCA()
{
    #varible from user input
    keystorePath=$1
    storepassValue=$2
    keypassValue=$3

    #generate ca keystore
    rm -rf $keystorePath/$CA_KEYSTORE_NAME
    $KEYTOOL -genkeypair -alias $CA_ALIAS -keystore $keystorePath/$CA_KEYSTORE_NAME -dname
$CA_DNAME -storepass $storepassValue -keypass $keypassValue -validity 3650 -keyalg $CA_KEYALG -
keysize 3072 -ext bc=ca:true
    if [ $? -ne 0 ]; then
        echo "generate ca.keystore failed."
        exit 1
    fi

    #generate ca.cer
    rm -rf "$keystorePath/ca.cer"
    $KEYTOOL -keystore "$keystorePath/$CA_KEYSTORE_NAME" -storepass "$storepassValue" -alias
```

```
$CA_ALIAS -validity 3650 -exportcert > "$keystorePath/ca.cer"
    if [ $? -ne 0 ]; then
        echo "generate ca.cer failed."
        exit 1
    fi

    #generate ca.truststore
    rm -rf "$keystorePath/flink.truststore"
    $KEYTOOL -importcert -keystore "$keystorePath/flink.truststore" -alias $CA_ALIAS -storepass
"$storepassValue" -noprompt -file "$keystorePath/ca.cer"
    if [ $? -ne 0 ]; then
        echo "generate ca.truststore failed."
        exit 1
    fi
}

function generateKeystore()
{
    #get path/pass from input
    keystorePath=$1
    storepassValue=$2
    keypassValue=$3

    #get value from conf
    aliasValue=$(getConf "flink.keystore.rsa.alias" "flink")
    validityValue=$(getConf "flink.keystore.rsa.validity" "3650")
    keyalgValue=$(getConf "flink.keystore.rsa.keyalg" "RSA")
    dnameValue=$(getConf "flink.keystore.rsa.dname" "CN=flink.com")
    SANValue=$(getConf "flink.keystore.rsa.ext" "ip:127.0.0.1")
    SANValue=$(echo "$SANValue" | xargs)
    SANValue="ip:$(echo "$SANValue"| sed 's/,/,ip:/g')"

    #generate keystore
    rm -rf $keystorePath/flink.keystore
    $KEYTOOL -genkeypair -alias $aliasValue -keystore $keystorePath/flink.keystore -dname $dnameValue -
ext SAN=$SANValue -storepass $storepassValue -keypass $keypassValue -keyalg $keyalgValue -keysize
3072 -validity 3650
    if [ $? -ne 0 ]; then
        echo "generate flink.keystore failed."
        exit 1
    fi

    #generate cer
    rm -rf $keystorePath/flink.csr
    $KEYTOOL -certreq -keystore $keystorePath/flink.keystore -storepass $storepassValue -alias $aliasValue -
file $keystorePath/flink.csr
    if [ $? -ne 0 ]; then
        echo "generate flink.csr failed."
        exit 1
    fi

    #generate flink.cer
    rm -rf $keystorePath/flink.cer
    $KEYTOOL -gencert -keystore $keystorePath/ca.keystore -storepass $storepassValue -alias $CA_ALIAS -
ext SAN=$SANValue -infile $keystorePath/flink.csr -outfile $keystorePath/flink.cer -validity 3650
    if [ $? -ne 0 ]; then
        echo "generate flink.cer failed."
        exit 1
    fi

    #import cer into keystore
    $KEYTOOL -importcert -keystore $keystorePath/flink.keystore -storepass $storepassValue -file
$keystorePath/ca.cer -alias $CA_ALIAS -noprompt
    if [ $? -ne 0 ]; then
        echo "importcert ca."
        exit 1
    fi

    $KEYTOOL -importcert -keystore $keystorePath/flink.keystore -storepass $storepassValue -file
```

```
$keystorePath/flink.cer -alias $aliasValue -noprompt;
    if [ $? -ne 0 ]; then
        echo "generate flink.truststore failed."
        exit 1
    fi
}

function configureFlinkConf()
{
    # set config
    if [ -f "$CLIENT_CONF_YAML" ]; then
        SSL_ENCRYPT_ENABLED=$(grep "security.ssl.encrypt.enabled" "$CLIENT_CONF_YAML" | awk '{print
$2}')
        if [ "$SSL_ENCRYPT_ENABLED" = "false" ];then

            sed -i s/"security.ssl.key-password:".*/"security.ssl.key-password:"\ "${keyPass}"/g
"$CLIENT_CONF_YAML"
            if [ $? -ne 0 ]; then
                echo "set security.ssl.key-password failed."
                return 1
            fi

            sed -i s/"security.ssl.keystore-password:".*/"security.ssl.keystore-password:"\ "${storePass}"/g
"$CLIENT_CONF_YAML"
            if [ $? -ne 0 ]; then
                echo "set security.ssl.keystore-password failed."
                return 1
            fi

            sed -i s/"security.ssl.truststore-password:".*/"security.ssl.truststore-password:"\ "${storePass}"/g
"$CLIENT_CONF_YAML"
            if [ $? -ne 0 ]; then
                echo "set security.ssl.keystore-password failed."
                return 1
            fi

            echo "security.ssl.encrypt.enabled is false, set security.ssl.key-password security.ssl.keystore-
password security.ssl.truststore-password success."
        else
            echo "security.ssl.encrypt.enabled is true, please enter security.ssl.key-password security.ssl.keystore-
password security.ssl.truststore-password encrypted value in flink-conf.yaml."
        fi

        keystoreFilePath="${keystorePath}"/flink.keystore
        sed -i 's#'"security.ssl.keystore:".*'#'"security.ssl.keystore:"\ "$keystoreFilePath"'#g'
"$CLIENT_CONF_YAML"
        if [ $? -ne 0 ]; then
            echo "set security.ssl.keystore failed."
            return 1
        fi


        truststoreFilePath="${keystorePath}/flink.truststore"
        sed -i 's#'"security.ssl.truststore:".*'#'"security.ssl.truststore:"\ "$truststoreFilePath"'#g'
"$CLIENT_CONF_YAML"
        if [ $? -ne 0 ]; then
            echo "set security.ssl.truststore failed."
            return 1
        fi

        command -v sha256sum >/dev/null
        if [ $? -ne 0 ];then
            echo "sha256sum is not exist, it will produce security.cookie with date +%F-%H-%M-%s-%N."
            cookie=$(date +%F-%H-%M-%s-%N)
        else
            cookie="$(echo "${KEYTABPRINCEPAL}"| sha256sum | awk '{print $1}')"
        fi

        sed -i s/"security.cookie:".*/"security.cookie:"\ "${cookie}"/g "$CLIENT_CONF_YAML"
```

```
        if [ $? -ne 0 ]; then
            echo "set security.cookie failed."
            return 1
        fi
    fi
    return 0;
}

main()
{
    #check environment variable is set or not
    if [ -z ${FLINK_HOME+x} ]; then
        echo "errro: environment variables are not set."
        exit 1
    fi
    stty -echo
    read -rp "Enter password:" password
    stty echo
    echo


    KEYTABPRINCEPAL=$(grep "security.kerberos.login.principal" "$CLIENT_CONF_YAML" | awk '{print $2}')
    if [ -z "$KEYTABPRINCEPAL" ];then
        echo "please config security.kerberos.login.principal info first."
        exit 1
    fi


    #get input
    keystorePath="$KEYSTOREPATH"
    storePass="$password"
    keyPass="$password"

    #generate self signed CA
    createSelfSignedCA "$keystorePath" "$storePass" "$keyPass"
    if [ $? -ne 0 ]; then
        echo "create self signed ca failed."
        exit 1
    fi

    #generate keystore
    generateKeystore "$keystorePath" "$storePass" "$keyPass"
    if [ $? -ne 0 ]; then
        echo "create keystore failed."
        exit 1
    fi

    echo "generate keystore/truststore success."

    # set flink config
    configureFlinkConf "$keystorePath" "$storePass" "$keyPass"
    if [ $? -ne 0 ]; then
        echo "configure Flink failed."
        exit 1
    fi

    return 0;
}

#the start main
main "$@"

exit 0
```

NOTE

Run the **sh generate_keystore.sh** *<password>* command. *<password>* is user-defined.

- If *<password>* contains the special character **$**, use the following method to avoid the password being escaped: **sh generate_keystore.sh 'Bigdata_2013'**.
- The password cannot contain **#**.
- Before using the **generate_keystore.sh** script, run the **source bigdata_env** command in the client directory.
- When the **generate_keystore.sh** script is used, the absolute paths of **security.ssl.keystore** and **security.ssl.truststore** are automatically filled in **flink-conf.yaml**. Therefore, you need to manually change the paths to relative paths as required. Example:
  - Change **/opt/client/Flink/flink/conf//flink.keystore** to **security.ssl.keystore: ssl/flink.keystore**.
  - Change **/opt/client/Flink/flink/conf//flink.truststore** to **security.ssl.truststore: ssl/flink.truststore**.
  - Create the **ssl** folder in any directory on the Flink client. For example, create the **ssl** folder in the **/opt/client/Flink/flink/conf/** directory and save the **flink.keystore** and **flink.truststore** files to the **ssl** folder.
  - When running the **yarn-session** or **flink run -m yarn-cluster** command, run the **yarn-session -t ssl -d** or **flink run -m yarn-cluster -yt ssl -d WordCount.jar** command in the same directory as the **ssl** folder.

12.7 Using Flume

12.7.1 Using Flume from Scratch

Scenario

You can use Flume to import collected log information to Kafka.

Prerequisites

- A streaming cluster with Kerberos authentication enabled has been created.
- The Flume client has been installed in a directory, for example, **/opt/Flumeclient**, on the node where logs are generated. For details about how to install the Flume client, see "Using Flume" > "Installing the Flume Client" in *MapReduce Service Component Operation Guide*. The client directory in the following operations is only an example. Change it to the actual installation directory.
- The streaming cluster can properly communicate with the node where logs are generated.

Using the Flume Client (Versions Earlier Than MRS 3.x)

NOTE

You do not need to perform **Step 2** to **Step 6** for a normal cluster.

Step 1 Install the client.

Step 2 Copy the configuration file of the authentication server from the Master1 node to the *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf** directory on the node where the Flume client resides.

The full file path is **${BIGDATA_HOME}/MRS_Current/1_X_KerberosClient/etc/kdc.conf**.

In the preceding paths, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 3 Check the service IP address of any node where the Flume role is deployed.

Log in to the cluster details page, choose *Name of the desired cluster* > **Components** > **Flume** > **Instances**, and check the service IP address of any node where the Flume role is deployed.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 4 Copy the user authentication file from this node to the *Flume client installation directory*/**fusioninsight-flume-Flume component version number/conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/MRS_XXX/install/FusionInsight-Flume-Flume component version number/flume/conf/flume.keytab**.

In the preceding paths, **XXX** indicates the product version number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 5 Copy the **jaas.conf** file from this node to the **conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/MRS_Current/1_X_Flume/etc/jaas.conf**.

In the preceding path, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 6 Log in to the Flume client node and go to the client installation directory. Run the following command to modify the file:

vi conf/jaas.conf

Change the full path of the user authentication file defined by **keyTab** to the **Flume client installation directory/fusioninsight-flume-*Flume component version number*/conf** saved in **Step 4**, and save the modification and exit.

Step 7 Run the following command to modify the **flume-env.sh** configuration file of the Flume client:

vi *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf/flume-env.sh**

Add the following information after **-XX:+UseCMSCompactAtFullCollection**:

```
-Djava.security.krb5.conf=Flume client installation directory/fusioninsight-flume-1.9.0/conf/kdc.conf -
Djava.security.auth.login.config=Flume client installation directory/fusioninsight-flume-1.9.0/conf/jaas.conf -
Dzookeeper.request.timeout=120000
```

For example, **"-XX:+UseCMSCompactAtFullCollection -Djava.security.krb5.conf=*Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/kdc.conf -Djava.security.auth.login.config=***Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/jaas.conf -Dzookeeper.request.timeout=120000"**

Change *Flume client installation directory* to the actual installation directory. Then save and exit.

Step 8 Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following command to restart the Flume client:

cd /opt/FlumeClient/fusioninsight-flume-*Flume component version number***/bin**

./flume-manage.sh restart

Step 9 Run the following command to modify the **properties.properties** configuration file of the Flume client:

vi *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf/properties.properties**

Add the following information to the file:

```
################################################################################
################
client.sources = static_log_source
client.channels = static_log_channel
client.sinks = kafka_sink
################################################################################
################
#LOG_TO_HDFS_ONLINE_1

client.sources.static_log_source.type = spooldir
client.sources.static_log_source.spoolDir = PATH
client.sources.static_log_source.fileSuffix = .COMPLETED
client.sources.static_log_source.ignorePattern = ^$
client.sources.static_log_source.trackerDir = PATH
client.sources.static_log_source.maxBlobLength = 16384
client.sources.static_log_source.batchSize = 51200
client.sources.static_log_source.inputCharset = UTF-8
client.sources.static_log_source.deserializer = LINE
client.sources.static_log_source.selector.type = replicating
client.sources.static_log_source.fileHeaderKey = file
client.sources.static_log_source.fileHeader = false
client.sources.static_log_source.basenameHeader = true
client.sources.static_log_source.basenameHeaderKey = basename
client.sources.static_log_source.deletePolicy = never

client.channels.static_log_channel.type = file
client.channels.static_log_channel.dataDirs = PATH
client.channels.static_log_channel.checkpointDir = PATH
client.channels.static_log_channel.maxFileSize = 2146435071
client.channels.static_log_channel.capacity = 1000000
client.channels.static_log_channel.transactionCapacity = 612000
client.channels.static_log_channel.minimumRequiredSpace = 524288000

client.sinks.kafka_sink.type = org.apache.flume.sink.kafka.KafkaSink
client.sinks.kafka_sink.kafka.topic = flume_test
client.sinks.kafka_sink.kafka.bootstrap.servers = XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka port number
```

```
client.sinks.kafka_sink.flumeBatchSize = 1000
client.sinks.kafka_sink.kafka.producer.type = sync
client.sinks.kafka_sink.kafka.security.protocol  = SASL_PLAINTEXT
client.sinks.kafka_sink.kafka.kerberos.domain.name = hadoop.XXX.com
client.sinks.kafka_sink.requiredAcks = 0

client.sources.static_log_source.channels = static_log_channel
client.sinks.kafka_sink.channel = static_log_channel
```

Modify the following parameters as required. Then save and exit the file.

- spoolDir
- trackerDir
- dataDirs
- checkpointDir
- topic

  If the topic does not exist in Kafka, the topic is automatically created by default.
- kafka.bootstrap.servers

  By default, the port for a security cluster is port 21007 and that for a normal cluster is port 9092.
- kafka.security.protocol

  Set this parameter to **SASL_PLAINTEXT** for a security cluster and **PLAINTEXT** for a normal cluster.
- **kafka.kerberos.domain.name**

  You do not need to set this parameter for a normal cluster. For a security cluster, the value of this parameter is the value of **kerberos.domain.name** in the Kafka cluster.

  You can check **${BIGDATA_HOME}/MRS_Current/1_X_Broker/etc/server.properties** on the node where the broker instance resides.

  In the preceding paths, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 10 The Flume client automatically loads the information in the **properties.properties** file.

After new log files are generated in the directory specified by **spoolDir**, the logs will be sent to Kafka producers and can be consumed by Kafka consumers.

----End

Using the Flume Client (MRS 3.x or Later)

NOTE

You do not need to perform **Step 2** to **Step 6** for a normal cluster.

Step 1 Install the client.

Step 2 Copy the configuration file of the authentication server from the Master1 node to the *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf** directory on the node where the Flume client resides.

The full file path is **${BIGDATA_HOME}/FusionInsight_Current/1_X_KerberosClient/etc/kdc.conf**. In the preceding path, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 3 Check the service IP address of any node where the Flume role is deployed.

Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster > Services > Flume > Instance**. Check the service IP address of any node where the Flume role is deployed.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 4 Copy the user authentication file from this node to the *Flume client installation directory*/**fusioninsight-flume-Flume component version number/conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/FusionInsight_Porter_XXX/install/FusionInsight-Flume-Flume component version number/flume/conf/flume.keytab**.

In the preceding paths, **XXX** indicates the product version number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 5 Copy the **jaas.conf** file from this node to the **conf** directory on the Flume client node.

The full file path is **${BIGDATA_HOME}/FusionInsight_Current/1_X_Flume/etc/jaas.conf**.

In the preceding path, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 6 Log in to the Flume client node and go to the client installation directory. Run the following command to modify the file:

vi conf/jaas.conf

Change the full path of the user authentication file defined by **keyTab** to the **Flume client installation directory/fusioninsight-flume-*Flume component version number*/conf** saved in **Step 4**, and save the modification and exit.

Step 7 Run the following command to modify the **flume-env.sh** configuration file of the Flume client:

vi *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf/flume-env.sh**

Add the following information after **-XX:+UseCMSCompactAtFullCollection**:

```
-Djava.security.krb5.conf=Flume client installation directory/fusioninsight-flume-1.9.0/conf/kdc.conf -Djava.security.auth.login.config=Flume client installation directory/fusioninsight-flume-1.9.0/conf/jaas.conf -Dzookeeper.request.timeout=120000
```

For example, **"-XX:+UseCMSCompactAtFullCollection -Djava.security.krb5.conf=*Flume client installation directory*/fusioninsight-flume-*Flume component version number*/conf/kdc.conf -Djava.security.auth.login.config=***Flume client installation directory***/fusioninsight-flume-***Flume component version number***/conf/jaas.conf -Dzookeeper.request.timeout=120000"**

Change *Flume client installation directory* to the actual installation directory. Then save and exit.

Step 8 Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following command to restart the Flume client:

cd /opt/FlumeClient/fusioninsight-flume-*Flume component version number***/bin**

./flume-manage.sh restart

Step 9 Run the following command to modify the **properties.properties** configuration file of the Flume client:

vi *Flume client installation directory*/**fusioninsight-flume-***Flume component version number*/**conf/properties.properties**

Add the following information to the file:

```
################################################################################################
client.sources = static_log_source
client.channels = static_log_channel
client.sinks = kafka_sink
################################################################################################
#LOG_TO_HDFS_ONLINE_1

client.sources.static_log_source.type = spooldir
client.sources.static_log_source.spoolDir = PATH
client.sources.static_log_source.fileSuffix = .COMPLETED
client.sources.static_log_source.ignorePattern = ^$
client.sources.static_log_source.trackerDir = PATH
client.sources.static_log_source.maxBlobLength = 16384
client.sources.static_log_source.batchSize = 51200
client.sources.static_log_source.inputCharset = UTF-8
client.sources.static_log_source.deserializer = LINE
client.sources.static_log_source.selector.type = replicating
client.sources.static_log_source.fileHeaderKey = file
client.sources.static_log_source.fileHeader = false
client.sources.static_log_source.basenameHeader = true
client.sources.static_log_source.basenameHeaderKey = basename
client.sources.static_log_source.deletePolicy = never

client.channels.static_log_channel.type = file
client.channels.static_log_channel.dataDirs = PATH
client.channels.static_log_channel.checkpointDir = PATH
client.channels.static_log_channel.maxFileSize = 2146435071
client.channels.static_log_channel.capacity = 1000000
client.channels.static_log_channel.transactionCapacity = 612000
client.channels.static_log_channel.minimumRequiredSpace = 524288000

client.sinks.kafka_sink.type = org.apache.flume.sink.kafka.KafkaSink
client.sinks.kafka_sink.kafka.topic = flume_test
client.sinks.kafka_sink.kafka.bootstrap.servers = XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka port number,XXX.XXX.XXX.XXX:Kafka port number
client.sinks.kafka_sink.flumeBatchSize = 1000
client.sinks.kafka_sink.kafka.producer.type = sync
client.sinks.kafka_sink.kafka.security.protocol  = SASL_PLAINTEXT
client.sinks.kafka_sink.kafka.kerberos.domain.name = hadoop.XXX.com
```

```
client.sinks.kafka_sink.requiredAcks = 0

client.sources.static_log_source.channels = static_log_channel
client.sinks.kafka_sink.channel = static_log_channel
```

Modify the following parameters as required. Then save and exit the file.

- spoolDir
- trackerDir
- dataDirs
- checkpointDir
- topic

  If the topic does not exist in Kafka, the topic is automatically created by default.
- kafka.bootstrap.servers

  By default, the port for a security cluster is port 21007 and that for a normal cluster is port 9092.
- kafka.security.protocol

  Set this parameter to **SASL_PLAINTEXT** for a security cluster and **PLAINTEXT** for a normal cluster.
- **kafka.kerberos.domain.name**

  You do not need to set this parameter for a normal cluster. For a security cluster, the value of this parameter is the value of **kerberos.domain.name** in the Kafka cluster.

  For details, check **${BIGDATA_HOME}/FusionInsight_Current/1_X_Broker/etc/server.properties** on the node where the broker instance resides.

  In the preceding paths, **X** indicates a random number. Change it based on the site requirements. The file must be saved by the user who installs the Flume client, for example, user **root**.

Step 10 The Flume client automatically loads the information in the **properties.properties** file.

After new log files are generated in the directory specified by **spoolDir**, the logs will be sent to Kafka producers and can be consumed by Kafka consumers.

----End

12.7.2 Overview

Flume is a distributed, reliable, and highly available system for aggregating massive logs, which can efficiently collect, aggregate, and move massive log data from different data sources and store the data in a centralized data storage system. Various data senders can be customized in the system to collect data. Additionally, Flume provides simple data processes capabilities and writes data to data receivers (which is customizable).

Flume consists of the client and server, both of which are FlumeAgents. The server corresponds to the FlumeServer instance and is directly deployed in a cluster. The client can be deployed inside or outside the cluster. he client-side and service-side FlumeAgents work independently and provide the same functions.

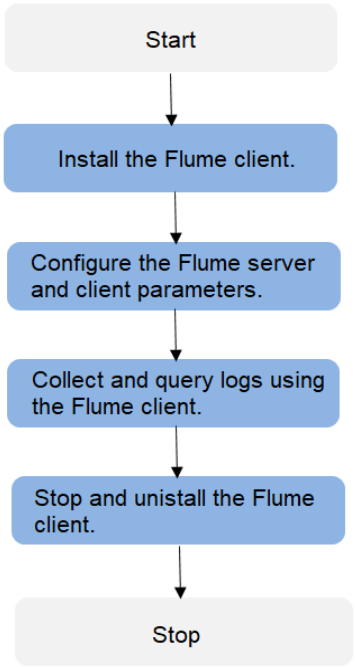
The client-side FlumeAgent needs to be independently installed. Data can be directly imported to components such as HDFS and Kafka. Additionally, the client-side and service-side FlumeAgents can also work together to provide services.

Process

The process for collecting logs using Flume is as follows:

1. Installing the flume client
2. Configuring the Flume server and client parameters
3. Collecting and querying logs using the Flume client
4. Stopping and uninstalling the Flume client

Figure 12-15 Log collection process



Flume Client

A Flume client consists of the source, channel, and sink. The source sends the data to the channel, and then the sink transmits the data from the channel to the external device. **Table 12-105** describes Flume modules.

Table 12-105 Module description

Name	Description
Source	A source receives or generates data and sends the data to one or multiple channels. The source can work in either data-driven or polling mode. Typical sources include: ● Sources that are integrated with the system and receives data, such as Syslog and Netcat ● Sources that automatically generate event data, such as Exec and SEQ ● IPC sources that are used for communication between agents, such as Avro A Source must associate with at least one channel.
Channel	A channel is used to buffer data between a source and a sink. After the sink transmits the data to the next channel or the destination, the cache is deleted automatically. The persistency of the channels varies with the channel types: ● Memory channel: non-persistency ● File channel: persistency implemented based on write-ahead logging (WAL) ● JDBC channel: persistency implemented based on the embedded database Channels support the transaction feature to ensure simple sequential operations. A channel can work with sources and sinks of any quantity.
Sink	Sink is responsible for sending data to the next hop or final destination and removing the data from the channel after successfully sending the data. Typical sinks include: ● Sinks that send storage data to the final destination, such as HDFS and Kafka ● Sinks that are consumed automatically, such as Null Sink ● IPC sinks that are used for communication between agents, such as Avro A sink must associate with at least one channel.

A Flume client can have multiple sources, channels, and sinks. A source can send data to multiple channels, and then multiple sinks send the data out of the client.

Multiple Flume clients can be cascaded. That is, a sink can send data to the source of another client.

Supplementary Information

1. Flume provides the following reliability measures:
   - The transaction mechanism is implemented between sources and channels, and between channels and sinks.
   - The sink processor supports the failover and load balancing (load_balance) mechanisms.

     The following is an example of the load balancing (load_balance) configuration:

     ```
     server.sinkgroups=g1
     server.sinkgroups.g1.sinks=k1 k2
     server.sinkgroups.g1.processor.type=load_balance
     server.sinkgroups.g1.processor.backoff=true
     server.sinkgroups.g1.processor.selector=random
     ```
2. The following are precautions for the aggregation and cascading of multiple Flume clients:
   - Avro or Thrift protocol can be used for cascading.
   - When the aggregation end contains multiple nodes, evenly distribute the clients to these nodes. Do not connect all the clients to a single node.
3. The Flume client can contain multiple independent data flows. That is, multiple sources, channels, and sinks can be configured in the **properties.properties** configuration file. These components can be linked to form multiple flows.

   For example, to configure two data flows in a configuration, run the following commands:

   ```
   server.sources = source1 source2
   server.sinks = sink1 sink2
   server.channels = channel1 channel2

   #dataflow1
   server.sources.source1.channels = channel1
   server.sinks.sink1.channel = channel1

   #dataflow2
   server.sources.source2.channels = channel2
   server.sinks.sink2.channel = channel2
   ```

12.7.3 Installing the Flume Client

12.7.3.1 Installing the Flume Client on Clusters of Versions Earlier Than MRS 3.x

Scenario

To use Flume to collect logs, you must install the Flume client on a log host. You can create an ECS and install the Flume client on it.

This section applies to versions earlier than MRS 3.x.

Prerequisites

- A streaming cluster with the Flume component has been created.
- The log host is in the same VPC and subnet with the MRS cluster.

- You have obtained the username and password for logging in to the log host.

Procedure

Step 1 Create an ECS that meets the requirements.

Step 2 Go to the cluster details page and choose **Components**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 3 Click **Download Client**.

1. In **Client Type**, select **All client files**.
2. In **Download to**, select **Remote host**.
3. Set **Host IP Address** to the IP address of the ECS, **Host Port** to **22**, and **Save Path** to **/tmp**.
   - If the default port **22** for logging in to an ECS through SSH has been changed, set **Host Port** to a new port.
   - The value of **Save Path** contains a maximum of 256 characters.
4. Set **Login User** to **root**.

   If another user is used, ensure that the user has permissions to read, write, and execute the save path.
5. Select **Password** or **SSH Private Key** for **Login Mode**.
   - **Password**: Enter the password of user **root** set during cluster creation.
   - **SSH Private Key**: Select and upload the key file used for creating the cluster.
6. Click **OK** to generate a client file.

   If the following information is displayed, the client package is saved.

   ```
   Client files downloaded to the remote host successfully.
   ```

   If the following information is displayed, check the username, password, and security group configurations of the remote host. Ensure that the username and password are correct and an inbound rule of the SSH (22) port has been added to the security group of the remote host. And then, go to **Step 3** to download the client again.

   ```
   Failed to connect to the server. Please check the network connection or parameter settings.
   ```

Step 4 Choose **Flume** > **Instance**. Query the **Business IP Address** of any Flume instance and any two MonitorServer instances.

Step 5 Log in to the ECS using VNC. See section "Login Using VNC" in the *Elastic Cloud Service User Guide* (**Instances** > **Logging In to a Linux ECS** > **Login Using VNC**.

All images support Cloud-Init. The preset username for Cloud-Init is **root** and the password is the one you set during cluster creation. You are advised to change the password upon the first login.

Step 6 On the ECS, switch to user **root** and copy the installation package to the **/opt** directory.

sudo su - root

cp /tmp/MRS_Flume_Client.tar /opt

Step 7 Run the following command in the **/opt** directory to decompress the package and obtain the verification file and the configuration package of the client:

tar -xvf MRS_Flume_Client.tar

Step 8 Run the following command to verify the configuration package of the client:

sha256sum -c MRS_Flume_ClientConfig.tar.sha256

If the following information is displayed, the file package is successfully verified:

```
MRS_Flume_ClientConfig.tar: OK
```

Step 9 Run the following command to decompress **MRS_Flume_ClientConfig.tar**:

tar -xvf MRS_Flume_ClientConfig.tar

Step 10 Run the following command to install the client running environment to a new directory, for example, **/opt/Flumeenv**. A directory is automatically generated during the client installation.

sh /opt/MRS_Flume_ClientConfig/install.sh /opt/Flumeenv

If the following information is displayed, the client running environment is successfully installed:

```
Components client installation is complete.
```

Step 11 Run the following command to configure environment variables:

source /opt/Flumeenv/bigdata_env

Step 12 Run the following commands to decompress the Flume client package:

cd /opt/MRS_Flume_ClientConfig/Flume

tar -xvf FusionInsight-Flume-1.6.0.tar.gz

Step 13 Run the following command to check whether the password of the current user has expired:

chage -l root

If the value of **Password expires** is earlier than the current time, the password has expired. Run the **chage -M -1 root** command to validate the password.

Step 14 Run the following command to install the Flume client to a new directory, for example, **/opt/FlumeClient**. A directory is automatically generated during the client installation.

sh /opt/MRS_Flume_ClientConfig/Flume/install.sh -d /opt/FlumeClient -f *service IP address of the MonitorServer instance* **-c** *path of the Flume configuration file* **-l /var/log/ -e** *service IP address of Flume* **-n** *name of the Flume client*

The parameters are described as follows:

- **-d**: indicates the installation path of the Flume client.

- (Optional) **-f**: indicates the service IP addresses of the two MonitorServer instances, separated by a comma (,). If the IP addresses are not configured, the Flume client will not send alarm information to MonitorServer, and the client information will not be displayed on MRS Manager.
- (Optional) **-c**: indicates the **properties.properties** configuration file that the Flume client loads after installation. If this parameter is not specified, the **fusioninsight-flume-1.6.0/conf/properties.properties** file in the client installation directory is used by default. The configuration file of the client is empty. You can modify the configuration file as required and the Flume client will load it automatically.
- (Optional) **-l**: indicates the log directory. The default value is **/var/log/Bigdata**.
- (Optional) **-e**: indicates the service IP address of the Flume instance. It is used to receive the monitoring indicators reported by the client.
- (Optional) **-n**: indicates the name of the Flume client.
- IBM JDK does not support **-Xloggc**. You must change **-Xloggc** to **-Xverbosegclog** in **flume/conf/flume-env.sh**. For 32-bit JDK, the value of **-Xmx** must not exceed 3.25 GB.
- In **flume/conf/flume-env.sh**, the default value of **-Xmx** is 4 GB. If the client memory is too small, you can change it to 512 MB or even 1 GB.

For example, run **sh install.sh -d /opt/FlumeClient**.

If the following information is displayed, the client is successfully installed:

```
install flume client successfully.
```

----End

12.7.3.2 Installing the Flume Client on Clusters of MRS 3.*x* or a Later Version

Scenario

To use Flume to collect logs, you must install the Flume client on a log host. You can create an ECS and install the Flume client on it.

This section applies to MRS 3.*x* and later versions.

Prerequisites

- A cluster with the Flume component has been created.
- The log host is in the same VPC and subnet with the MRS cluster.
- You have obtained the username and password for logging in to the log host.
- The installation directory is automatically created if it does not exist. If it exists, the directory must be left blank. The directory path cannot contain any space.

Procedure

Step 1 Obtain the software package.

Log in to the FusionInsight Manager. Choose **Cluster** > *Name of the target cluster* > **Services** > **Flume**. On the Flume service page that is displayed, choose **More** >

Download Client in the upper right corner and set **Select Client Type** to **Complete Client** to download the Flume service client file.

The file name of the client is **FusionInsight_Cluster**_<*Cluster ID*>**_Flume_Client.tar**. This section takes the client file **FusionInsight_Cluster_1_Flume_Client.tar** as an example.

Step 2 Upload the software package.

Upload the software package to a directory, for example, **/opt/client** on the node where the Flume service client will be installed as user **user**.

NOTE

user is the user who installs and runs the Flume client.

Step 3 Decompress the software package.

Log in to the node where the Flume service client is to be installed as user **user**. Go to the directory where the installation package is installed, for example, **/opt/client**, and run the following command to decompress the installation package to the current directory:

cd /opt/client

tar -xvf FusionInsight_Cluster_1_Flume_Client.tar

Step 4 Verify the software package.

Run the **sha256sum -c** command to verify the decompressed file. If **OK** is returned, the verification is successful. Example:

sha256sum -c FusionInsight_Cluster_1_Flume_ClientConfig.tar.sha256

```
FusionInsight_Cluster_1_Flume_ClientConfig.tar: OK
```

Step 5 Decompress the package.

tar -xvf FusionInsight_Cluster_1_Flume_ClientConfig.tar

Step 6 Run the following command in the Flume client installation directory to install the client to a specified directory (for example, **opt/FlumeClient**): After the client is installed successfully, the installation is complete.

cd /opt/client/FusionInsight_Cluster_1_Flume_ClientConfig/Flume/FlumeClient

./install.sh -d */opt/FlumeClient* **-f** *MonitorServerService IP address or host name of the role* **-c** *User service configuration filePath for storing properties.properties* **-s** *CPU threshold* **-l /var/log/Bigdata -e** *FlumeServer service IP address or host name* **-n** *Flume*

NOTE

- **-d**: Flume client installation path
- (Optional) **-f**: IP addresses or host names of two MonitorServer roles. The IP addresses or host names are separated by commas (,). If this parameter is not configured, the Flume client does not send alarm information to MonitorServer and information about the client cannot be viewed on the FusionInsight Manager GUI.
- (Optional) **-c**: Service configuration file, which needs to be generated by the user based on the service. For details about how to generate the file on the configuration tool page of the Flume server, see **Flume Service Configuration Guide**. Upload the file to any directory on the node where the client is to be installed. If this parameter is not specified during the installation, you can upload the generated service configuration file **properties.properties** to the **/opt/FlumeClient/fusioninsight-flume-1.9.0/conf** directory after the installation.
- (Optional) **-s**: cgroup threshold. The value is an integer ranging from 1 to 100 x *N*. *N* indicates the number of CPU cores. The default threshold is **-1**, indicating that the processes added to the cgroup are not restricted by the CPU usage.
- (Optional) **-l**: Log path. The default value is **/var/log/Bigdata**. The user **user** must have the write permission on the directory. When the client is installed for the first time, a subdirectory named **flume-client** is generated. After the installation, subdirectories named **flume-client-***n* will be generated in sequence. The letter *n* indicates a sequence number, which starts from 1 in ascending order. In the **/conf/** directory of the Flume client installation directory, modify the **ENV_VARS** file and search for the **FLUME_LOG_DIR** attribute to view the client log path.
- (Optional) **-e**: Service IP address or host name of FlumeServer, which is used to receive statistics for the monitoring indicator reported by the client.
- (Optional) **-n**: Name of the Flume client. You can choose **Cluster** > *Name of the desired cluster* > **Service** > **Flume** > **Flume Management** on FusionInsight Manager to view the client name on the corresponding node.
- If the following error message is displayed, run the **export JAVA_HOME=*JDK path*** command.
  JAVA_HOME is null in current user,please install the JDK and set the JAVA_HOME
- IBM JDK does not support **-Xloggc**. You must change **-Xloggc** to **-Xverbosegclog** in **flume/conf/flume-env.sh**. For 32-bit JDK, the value of **-Xmx** must not exceed 3.25 GB.
- When installing a cross-platform client in a cluster, go to the **/opt/client/FusionInsight_Cluster_1_Flume_ClientConfig/Flume/FusionInsight-Flume-1.9.0.tar.gz** directory to install the Flume client.

----End

12.7.4 Viewing Flume Client Logs

Scenario

You can view logs to locate faults.

Prerequisites

The Flume client has been installed.

Procedure

Step 1 Go to the Flume client log directory (**/var/log/Bigdata** by default).

Step 2 Run the following command to view the log file:

ls -lR flume-client-*

A log file is shown as follows:

```
flume-client-1/flume:
total 7672
-rw-------. 1 root root       0 Sep  8 19:43 Flume-audit.log
-rw-------. 1 root root 1562037 Sep 11 06:05 FlumeClient.2017-09-11_04-05-09.[1].log.zip
-rw-------. 1 root root 6127274 Sep 11 14:47 FlumeClient.log
-rw-------. 1 root root    2935 Sep  8 22:20 flume-root-20170908202009-pid72456-gc.log.0.current
-rw-------. 1 root root    2935 Sep  8 22:27 flume-root-20170908202634-pid78789-gc.log.0.current
-rw-------. 1 root root    4382 Sep  8 22:47 flume-root-20170908203137-pid84925-gc.log.0.current
-rw-------. 1 root root    4390 Sep  8 23:46 flume-root-20170908204918-pid103920-gc.log.0.current
-rw-------. 1 root root    3196 Sep  9 10:12 flume-root-20170908215351-pid44372-gc.log.0.current
-rw-------. 1 root root    2935 Sep  9 10:13 flume-root-20170909101233-pid55119-gc.log.0.current
-rw-------. 1 root root    6441 Sep  9 11:10 flume-root-20170909101631-pid59301-gc.log.0.current
-rw-------. 1 root root       0 Sep  9 11:10 flume-root-20170909111009-pid119477-gc.log.0.current
-rw-------. 1 root root   92896 Sep 11 13:24 flume-root-20170909111126-pid120689-gc.log.0.current
-rw-------. 1 root root    5588 Sep 11 14:46 flume-root-20170911132445-pid42259-gc.log.0.current
-rw-------. 1 root root    2576 Sep 11 13:24 prestartDetail.log
-rw-------. 1 root root    3303 Sep 11 13:24 startDetail.log
-rw-------. 1 root root    1253 Sep 11 13:24 stopDetail.log

flume-client-1/monitor:
total 8
-rw-------. 1 root root  141 Sep  8 19:43 flumeMonitorChecker.log
-rw-------. 1 root root 2946 Sep 11 13:24 flumeMonitor.log
```

In the log file, **FlumeClient.log** is the run log of the Flume client.

----End

12.7.5 Stopping or Uninstalling the Flume Client

Scenario

You can stop and start the Flume client or uninstall the Flume client when the Flume data ingestion channel is not required.

Procedure

- Stop the Flume client of the Flume role.

  Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following command to stop the Flume client:

  cd /opt/FlumeClient/fusioninsight-flume-*Flume component version number***/bin**

  ./flume-manage.sh stop

  If the following information is displayed after the command execution, the Flume client is successfully stopped.

  ```
  Stop Flume PID=120689 successful..
  ```

  NOTE

  The Flume client will be automatically restarted after being stopped. If you do not need automatic restart, run the following command:

  ./flume-manage.sh stop force

  If you want to restart the Flume client, run the following command:

  ./flume-manage.sh start force

- Uninstall the Flume client of the Flume role.

  Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following command to uninstall the Flume client:

  cd /opt/FlumeClient/fusioninsight-flume-*Flume component version number***/inst**

  ./uninstall.sh

12.7.6 Using the Encryption Tool of the Flume Client

Scenario

You can use the encryption tool provided by the Flume client to encrypt some parameter values in the configuration file.

Prerequisites

The Flume client has been installed.

Procedure

Step 1 Log in to the Flume client node and go to the client installation directory, for example, **/opt/FlumeClient**.

Step 2 Run the following command to switch the directory:

cd fusioninsight-flume-*Flume component version number***/bin**

Step 3 Run the following command to encrypt information:

./genPwFile.sh

Input the information that you want to encrypt twice.

Step 4 Run the following command to query the encrypted information:

cat password.property

NOTE

If the encryption parameter is used for the Flume server, you need to perform encryption on the corresponding Flume server node. You need to run the encryption script as user **omm** for encryption.

- For versions earlier than MRS 3.x, the encryption path is **/opt/Bigdata/MRS_XXX/install/FusionInsight-Flume-*Flume component version number*/flume/bin/genPwFile.sh**.
- For MRS 3.*x* or later, the encryption path is **/opt/Bigdata/FusionInsight_Porter_XXX/install/FusionInsight-Flume-*Flume component version number*/flume/bin/genPwFile.sh**. *XXX* indicates the product version number.

----End

12.7.7 Flume Service Configuration Guide

This section applies to MRS 3.*x* and later versions.

This configuration guide describes how to configure common Flume services. For non-common Source, Channel, and Sink configuration, see the user manual

provided by the Flume community. You can obtain the user manual at **http://flume.apache.org/releases/1.9.0.html**.

NOTE

- Parameters in bold in the following tables are mandatory.
- The value of **BatchSize** of the Sink must be less than that of **transactionCapacity** of the Channel.
- Only some parameters of Source, Channel, and Sink are displayed on the Flume configuration tool page. For details, see the following configurations.
- The Customer Source, Customer Channel, and Customer Sink displayed on the Flume configuration tool page need to be configured based on self-developed code. The following common configurations are not displayed.

Common Source Configurations

- **Avro Source**

  An Avro source listens to the Avro port, receives data from the external Avro client, and places data into configured channels. Common configurations are as follows:

  Table 12-106 Common configurations of an Avro source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured.
type	avro	Specifies the type of the avro source, which must be **avro**.
bind	-	Specifies the listening host name/IP address.
port	-	Specifies the bound listening port. Ensure that this port is not occupied.
threads	-	Specifies the maximum number of source threads.
compression-type	none	Specifies the message compression format, which can be set to **none** or **deflate**. **none** indicates that data is not compressed, while **deflate** indicates that data is compressed.

Parameter	Default Value	Description
compression-level	6	Specifies the data compression level, which ranges from **1** to **9**. The larger the value is, the higher the compression rate is.
ssl	false	Specifies whether to use SSL encryption. If this parameter is set to **true**, the values of **keystore** and **keystore-password** must be specified.
truststore-type	JKS	Specifies the Java trust store type, which can be set to **JKS** or **PKCS12**. **NOTE** Different passwords are used to protect the key store and private key of **JKS**, while the same password is used to protect the key store and private key of **PKCS12**.
truststore	-	Specifies the Java trust store file.
truststore-password	-	Specifies the Java trust store password.
keystore-type	JKS	Specifies the keystore type set after SSL is enabled, which can be set to **JKS** or **PKCS12**. **NOTE** Different passwords are used to protect the key store and private key of **JKS**, while the same password is used to protect the key store and private key of **PKCS12**.
keystore	-	Specifies the keystore file path set after SSL is enabled. This parameter is mandatory if SSL is enabled.

Parameter	Default Value	Description
keystore-password	-	Specifies the keystore password set after SSL is enabled. This parameter is mandatory if SSL is enabled.
trust-all-certs	false	Specifies whether to disable the check for the SSL server certificate. If this parameter is set to **true**, the SSL server certificate of the remote source is not checked. You are not advised to perform this operation during the production.
exclude-protocols	SSLv3	Specifies the excluded protocols. The entered protocols must be separated by spaces. The default value is **SSLv3**.
ipFilter	false	Specifies whether to enable the IP address filtering.
ipFilter.rules	-	Specifies the rules of *N* network **ipFilters**. Host names or IP addresses must be separated by commas (,). If this parameter is set to **true**, there are two configuration rules: allow and forbidden. The configuration format is as follows: ipFilterRules=allow:ip:127.*, allow:name:localhost, deny:ip:*

- **SpoolDir Source**

  SpoolDir Source monitors and transmits new files that have been added to directories in real-time mode. Common configurations are as follows:

Table 12-107 Common configurations of a Spooling Directory source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured.
type	spooldir	Specifies the type of the spooling source, which must be set to **spooldir**.
spoolDir	-	Specifies the monitoring directory of the Spooldir source. A Flume running user must have the read, write, and execution permissions on the directory.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the source is restarted. Unit: second
fileSuffix	.COMPLETED	Specifies the suffix added after file transmission is complete.
deletePolicy	never	Specifies the source file deletion policy after file transmission is complete. The value can be either **never** or **immediate**. **never** indicates that the source file is not deleted after file transmission is complete, while **immediate** indicates that the source file is immediately deleted after file transmission is complete.
ignorePattern	^$	Specifies the regular expression of a file to be ignored. The default value is ^$, indicating that spaces are ignored.
includePattern	^.*$	Specifies the regular expression that contains a file. This parameter can be used together with **ignorePattern**. If a file meets both **ignorePattern** and **includePattern**, the file is ignored. In addition, when a file starts with a period (.), the file will not be filtered.
trackerDir	.flumespool	Specifies the metadata storage path during data transmission.
batchSize	1000	Specifies the number of events written to the channel in batches.
decodeErrorPolicy	FAIL	Specifies the code error policy. **NOTE** If a code error occurs in the file, set **decodeErrorPolicy** to **REPLACE** or **IGNORE**. Flume will skip the code error and continue to collect subsequent logs.

Parameter	Default Value	Description
deserializer	LINE	Specifies the file parser. The value can be either **LINE** or **BufferedLine**. • When the value is set to **LINE**, characters read from the file are transcoded one by one. • When the value is set to **BufferedLine**, one line or multiple lines of characters read from the file are transcoded in batches, which delivers better performance.
deserializer.maxLineLength	2048	Specifies the maximum length for resolution by line.
deserializer.maxBatchLine	1	Specifies the maximum number of lines for resolution by line. If multiple lines are set, **maxLineLength** must be set to a corresponding multiplier. **NOTE** When configuring the Interceptor, take the multi-line combination into consideration to avoid data loss. If the Interceptor cannot process combined lines, set this parameter to **1**.
selector.type	replicating	Specifies the selector type. The value can be either **replicating** or **multiplexing**. **replicating** indicates that data is replicated and then transferred to each channel so that each channel receives the same data, while **multiplexing** indicates that a channel is selected based on the value of the header in the event and each channel has different data.
interceptors	-	Specifies the interceptor. Multiple interceptors are separated by spaces.
inputCharset	UTF-8	Specifies the encoding format of a read file. The encoding format must be the same as that of the data source file that has been read. Otherwise, an error may occur during character parsing.
fileHeader	false	Specifies whether to add the file name (including the file path) to the event header.

Parameter	Default Value	Description
fileHeaderKey	-	Specifies that the data storage structure in header is set in the <key,value> mode. Parameters **fileHeaderKey** and **fileHeader** must be used together. Following is an example if **fileHeader** is set to true: Define **fileHeaderKey** as **file**. When the **/root/a.txt** file is read, **fileHeaderKey** exists in the header in the **file=/root/a.txt** format.
basenameHeader	false	Specifies whether to add the file name (excluding the file path) to the event header.
basenameHeaderKey	-	Specifies that the data storage structure in header is set in the <key,value> mode. Parameters **basenameHeaderKey** and **basenameHeader** must be used together. Following is an example if **basenameHeader** is set to **true**: Define **basenameHeaderKey** as **file**. When the **a.txt** file is read, **fileHeaderKey** exists in the header in the **file=a.txt** format.
pollDelay	500	Specifies the delay for polling new files in the monitoring directory. Unit: milliseconds
recursiveDirectorySearch	false	Specifies whether to monitor new files in the subdirectory of the configured directory.
consumeOrder	oldest	Specifies the consumption order of files in a directory. If this parameter is set to **oldest** or **youngest**, the sequence of files to be read is determined by the last modification time of files in the monitored directory. If there are a large number of files in the directory, it takes a long time to search for **oldest** or **youngest** files. If this parameter is set to **random**, an earlier created file may not be read for a long time. If this parameter is set to **oldest** or **youngest**, it takes a long time to find the latest and the earliest file. The options are as follows: **random**, **youngest**, and **oldest**.

Parameter	Default Value	Description
maxBackoff	4000	Specifies the maximum time to wait between consecutive attempts to write to a channel if the channel is full. If the time exceeds the threshold, an exception is thrown. The corresponding source starts to write at a smaller time value. Each time the source attempts, the digital exponent increases until the current specified value is reached. If data cannot be written, the data write fails. Unit: second
emptyFileEvent	true	Specifies whether to collect empty file information and send it to the sink end. The default value is **true**, indicating that empty file information is sent to the sink end. This parameter is valid only for HDFS Sink. Taking HDFS Sink as an example, if this parameter is set to **true** and an empty file exists in the **spoolDir** directory, an empty file with the same name will be created in the **hdfs.path** directory of HDFS.

NOTE

SpoolDir Source ignores the last line feed character of each event when data is reading by row. Therefore, Flume does not calculate the data volume counters used by the last line feed character.

- **Kafka Source**

  A Kafka source consumes data from Kafka topics. Multiple sources can consume data of the same topic, and the sources consume different partitions of the topic. Common configurations are as follows:

  Table 12-108 Common configurations of a Kafka source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured.
type	org.apache.flume.source.kafka.KafkaSource	Specifies the type of the Kafka source, which must be set to **org.apache.flume.source.kafka.KafkaSource**.

Parameter	Default Value	Description
kafka.bootstrap.servers	-	Specifies the bootstrap address port list of Kafka. If Kafka has been installed in the cluster and the configuration has been synchronized to the server, you do not need to set this parameter on the server. The default value is the list of all brokers in the Kafka cluster. This parameter must be configured on the client. Use commas (,) to separate multiple values of *IP address:Port number*. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT).
kafka.topics	-	Specifies the list of subscribed Kafka topics, which are separated by commas (,).
kafka.topics.regex	-	Specifies the subscribed topics that comply with regular expressions. **kafka.topics.regex** has a higher priority than **kafka.topics** and will overwrite **kafka.topics**.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the source is restarted. Unit: second
nodatatime	0 (Disabled)	Specifies the alarm threshold. An alarm is triggered when the duration that Kafka does not release data to subscribers exceeds the threshold. Unit: second This parameter can be configured in the **properties.properties** file.
batchSize	1000	Specifies the number of events written to the channel in batches.
batchDurationMillis	1000	Specifies the maximum duration of topic data consumption at a time, expressed in milliseconds.
keepTopicInHeader	false	Specifies whether to save topics in the event header. If the parameter value is **true**, topics configured in Kafka Sink become invalid.

Parameter	Default Value	Description
setTopicHeader	true	If this parameter is set to **true**, the topic name defined in **topicHeader** is stored in the header.
topicHeader	topic	When **setTopicHeader** is set to **true**, this parameter specifies the name of the topic received by the storage device. If the property is used with that of Kafka Sink **topicHeader**, be careful not to send messages to the same topic cyclically.
useFlumeEventFormat	false	By default, an event is transferred from a Kafka topic to the body of the event in the form of bytes. If this parameter is set to **true**, the Avro binary format of Flume is used to read events. When used together with the **parseAsFlumeEvent** parameter with the same name in KafkaSink or KakfaChannel, any set **header** generated from the data source is retained.
keepPartitionInHeader	false	Specifies whether to save partition IDs in the event header. If the parameter value is **true**, Kafka Sink writes data to the corresponding partition.
kafka.consumer.group.id	flume	Specifies the Kafka consumer group ID. Sources or proxies having the same ID are in the same consumer group.
kafka.security.protocol	SASL_PLAINTEXT	Specifies the Kafka security protocol. The parameter value must be set to PLAINTEXT in a common cluster. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT).

Parameter	Default Value	Description
Other Kafka Consumer Properties	-	Specifies other Kafka configurations. This parameter can be set to any consumption configuration supported by Kafka, and the **.kafka** prefix must be added to the configuration.

- **Taildir Source**

  A Taildir source monitors file changes in a directory and automatically reads the file content. In addition, it can transmit data in real time. Common configurations are as follows:

  Table 12-109 Common configurations of a Taildir source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured.
type	TAILDIR	Specifies the type of the taildir source, which must be set to TAILDIR.
filegroups	-	Specifies the group name of a collection file directory. Group names are separated by spaces.
filegroups.<filegroupName>.parentDir	-	Specifies the parent directory. The value must be an absolute path.
filegroups.<filegroupName>.filePattern	-	Specifies the relative file path of the file group's parent directory. Directories can be included and regular expressions are supported. It must be used together with **parentDir**.
positionFile	-	Specifies the metadata storage path during data transmission.
headers.<filegroupName>.<headerKey>	-	Specifies the key-value of an event when data of a group is being collected.
byteOffsetHeader	false	Specifies whether each event header contains the event location information in the source file. If the parameter value is true, the location information is saved in the byteoffset variable.

Parameter	Default Value	Description
maxBatchCount	Long.MAX_VALUE	Specifies the maximum number of batches that can be consecutively read from a file. If the monitored directory reads multiple files consecutively and one of the files is written at a rapid rate, other files may fail to be processed. This is because the file that is written at a high speed will be in an infinite read loop. In this case, set this parameter to a smaller value.
skipToEnd	false	Specifies whether Flume can locate the latest location of a file and read the latest data after restart. If the parameter value is true, Flume locates and reads the latest file data after restart.
idleTimeout	120000	Specifies the idle duration during file reading, expressed in milliseconds. If file content is not changed in the preset time duration, close the file. If data is written to this file after the file is closed, open the file and read data.
writePosInterval	3000	Specifies the interval for writing metadata to a file, expressed in milliseconds.
batchSize	1000	Specifies the number of events written to the channel in batches.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the source is restarted. Unit: second
fileHeader	false	Specifies whether to add the file name (including the file path) to the event header.
fileHeaderKey	file	Specifies that the data storage structure in header is set in the <key,value> mode. Parameters **fileHeaderKey** and **fileHeader** must be used together. Following is an example if **fileHeader** is set to true: Define **fileHeaderKey** as **file**. When the **/root/a.txt** file is read, **fileHeaderKey** exists in the header in the **file=/root/a.txt** format.

- **Http Source**

  An HTTP source receives data from an external HTTP client and sends the data to the configured channels. Common configurations are as follows:

  Table 12-110 Common configurations of an HTTP source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured.
type	http	Specifies the type of the http source, which must be set to http.
bind	-	Specifies the listening host name/IP address.
port	-	Specifies the bound listening port. Ensure that this port is not occupied.
handler	org.apache.flume.source.http.JSONHandler	Specifies the message parsing method of an HTTP request. Two formats are supported: JSON (org.apache.flume.source.http.JSONHandler) and BLOB (org.apache.flume.sink.solr.morphline.BlobHandler).
handler.*	-	Specifies handler parameters.
exclude-protocols	SSLv3	Specifies the excluded protocols. The entered protocols must be separated by spaces. The default value is **SSLv3**.
include-cipher-suites	-	Specifies the included protocols. The entered protocols must be separated by spaces. If this parameter is left empty, all protocols are supported by default.
enableSSL	false	Specifies whether SSL is enabled in HTTP. If this parameter is set to **true**, the values of **keystore** and **keystore-password** must be specified.
keystore-type	JKS	Specifies the keystore type, which can be **JKS** or **PKCS12**.

Parameter	Default Value	Description
keystore	-	Specifies the keystore path set after SSL is enabled in HTTP.
keystorePassword	-	Specifies the keystore password set after SSL is enabled in HTTP.

- **Thrift Source**

  Thrift Source monitors the thrift port, receives data from the external Thrift clients, and puts the data into the configured channel. Common configurations are as follows:

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured.
type	thrift	Specifies the type of the thrift source, which must be set to **thrift**.
bind	-	Specifies the listening host name/IP address.
port	-	Specifies the bound listening port. Ensure that this port is not occupied.
threads	-	Specifies the maximum number of worker threads that can be run.
kerberos	false	Specifies whether Kerberos authentication is enabled.
agent-keytab	-	Specifies the address of the keytab file used by the server. The machine-machine account must be used. You are advised to use **flume/conf/flume_server.keytab** in the Flume service installation directory.
agent-principal	-	Specifies the principal of the security user used by the server. The principal must be a machine-machine account. You are advised to use the default user of Flume: flume_server/hadoop.*<system domain name>*@*<system domain name>* **NOTE** **flume_server/hadoop.**<*system domain name*> is the username. All letters in the system domain name contained in the username are lowercase letters. For example, **Local Domain** is set to **9427068F-6EFA-4833-B43E-60CB641E5B6C.COM**, and the username is **flume_server/hadoop.9427068f-6efa-4833-b43e-60cb641e5b6c.com**.

Parameter	Default Value	Description
compression-type	none	Specifies the message compression format, which can be set to **none** or **deflate**. **none** indicates that data is not compressed, while **deflate** indicates that data is compressed.
ssl	false	Specifies whether to use SSL encryption. If this parameter is set to **true**, the values of **keystore** and **keystore-password** must be specified.
keystore-type	JKS	Specifies the keystore type set after SSL is enabled.
keystore	-	Specifies the keystore file path set after SSL is enabled. This parameter is mandatory if SSL is enabled.
keystore-password	-	Specifies the keystore password set after SSL is enabled. This parameter is mandatory if SSL is enabled.

Common Channel Configurations

- **Memory Channel**

  A memory channel uses memory as the cache. Events are stored in memory queues. Common configurations are as follows:

  Table 12-111 Common configurations of a memory channel

Parameter	Default Value	Description
type	-	Specifies the type of the memory channel, which must be set to **memory**.
capacity	10000	Specifies the maximum number of events cached in a channel.

Parameter	Default Value	Description
transactionCapacity	1000	Specifies the maximum number of events accessed each time. **NOTE** • The parameter value must be greater than the batchSize of the source and sink. • The value of **transactionCapacity** must be less than or equal to that of **capacity**.
channelfullcount	10	Specifies the channel full count. When the count reaches the threshold, an alarm is reported.
keep-alive	3	Specifies the waiting time of the Put and Take threads when the transaction or channel cache is full. Unit: second
byteCapacity	80% of the maximum JVM memory	Specifies the total bytes of all event bodies in a channel. The default value is the 80% of the maximum JVM memory (indicated by **-Xmx**). Unit: bytes
byteCapacityBufferPercentage	20	Specifies the percentage of bytes in a channel (%).

- **File Channel**

  A file channel uses local disks as the cache. Events are stored in the folder specified by **dataDirs**. Common configurations are as follows:

  Table 12-112 Common configurations of a file channel

Parameter	Default Value	Description
type	-	Specifies the type of the file channel, which must be set to **file**.

Parameter	Default Value	Description
checkpointDir	${BIGDATA_DATA_HOME}/hadoop/data1~N/flume/checkpoint **NOTE** This path is changed with the custom data path.	Specifies the checkpoint storage directory.
dataDirs	${BIGDATA_DATA_HOME}/hadoop/data1~N/flume/data **NOTE** This path is changed with the custom data path.	Specifies the data cache directory. Multiple directories can be configured to improve performance. The directories are separated by commas (,).
maxFileSize	2146435071	Specifies the maximum size of a single cache file, expressed in bytes.
minimumRequiredSpace	524288000	Specifies the minimum idle space in the cache, expressed in bytes.
capacity	1000000	Specifies the maximum number of events cached in a channel.
transactionCapacity	10000	Specifies the maximum number of events accessed each time. **NOTE** • The parameter value must be greater than the batchSize of the source and sink. • The value of **transactionCapacity** must be less than or equal to that of **capacity**.
channelfullcount	10	Specifies the channel full count. When the count reaches the threshold, an alarm is reported.

Parameter	Default Value	Description
useDualCheckpoints	false	Specifies the backup checkpoint. If this parameter is set to **true**, the **backupCheckpointDir** parameter value must be set.
backupCheckpointDir	-	Specifies the path of the backup checkpoint.
checkpointInterval	30000	Specifies the check interval, expressed in seconds.
keep-alive	3	Specifies the waiting time of the Put and Take threads when the transaction or channel cache is full. Unit: second
use-log-replay-v1	false	Specifies whether to enable the old reply logic.
use-fast-replay	false	Specifies whether to enable the queue reply.
checkpointOnClose	true	Specifies that whether a checkpoint is created when a channel is disabled.

- **Memory File Channel**

  A memory file channel uses both memory and local disks as its cache and supports message persistence. It provides similar performance as a memory channel and better performance than a file channel. This channel is currently experimental and not recommended for use in production. The following table describes common configuration items: Common configurations are as follows:

  Table 12-113 Common configurations of a memory file channel

Parameter	Default Value	Description
type	org.apache.flume.channel.MemoryFileChannel	Specifies the type of the memory file channel, which must be set to **org.apache.flume.channel.MemoryFileChannel**.

Parameter	Default Value	Description
capacity	50000	Specifies the maximum number of events cached in a channel.
transactionCapacity	5000	Specifies the maximum number of events processed by a transaction. **NOTE** • The parameter value must be greater than the batchSize of the source and sink. • The value of **transactionCapacity** must be less than or equal to that of **capacity**.
subqueueByteCapacity	20971520	Specifies the maximum size of events that can be stored in a subqueue, expressed in bytes. A memory file channel uses both queues and subqueues to cache data. Events are stored in a subqueue, and subqueues are stored in a queue. **subqueueCapacity** and **subqueueInterval** determine the size of events that can be stored in a subqueue. **subqueueCapacity** specifies the capacity of a subqueue, and **subqueueInterval** specifies the duration that a subqueue can store events. Events in a subqueue are sent to the destination only after the subqueue reaches the upper limit of **subqueueCapacity** or **subqueueInterval**. **NOTE** The value of **subqueueByteCapacity** must be greater than the number of events specified by **batchSize**.
subqueueInterval	2000	Specifies the maximum duration that a subqueue can store events, expressed in milliseconds.
keep-alive	3	Specifies the waiting time of the Put and Take threads when the transaction or channel cache is full. Unit: second
dataDir	-	Specifies the cache directory for local files.
byteCapacity	80% of the maximum JVM memory	Specifies the channel cache capacity. Unit: bytes

Parameter	Default Value	Description
compression-type	None	Specifies the message compression format, which can be set to **none** or **deflate**. **none** indicates that data is not compressed, while **deflate** indicates that data is compressed.
channelfullcount	10	Specifies the channel full count. When the count reaches the threshold, an alarm is reported.

The following is a configuration example of a memory file channel:

```
server.channels.c1.type = org.apache.flume.channel.MemoryFileChannel
server.channels.c1.dataDir = /opt/flume/mfdata
server.channels.c1.subqueueByteCapacity = 20971520
server.channels.c1.subqueueInterval=2000
server.channels.c1.capacity = 500000
server.channels.c1.transactionCapacity = 40000
```

- **Kafka Channel**

  A Kafka channel uses a Kafka cluster as the cache. Kafka provides high availability and multiple copies to prevent data from being immediately consumed by sinks when Flume or Kafka Broker crashes.

  Table 12-114 Common configurations of a Kafka channel

Parameter	Default Value	Description
type	-	Specifies the type of the Kafka channel, which must be set to **org.apache.flume.channel.kafka.KafkaChannel**.

Parameter	Default Value	Description
kafka.bootstrap.servers	-	Specifies the bootstrap address port list of Kafka. If Kafka has been installed in the cluster and the configuration has been synchronized to the server, you do not need to set this parameter on the server. The default value is the list of all brokers in the Kafka cluster. This parameter must be configured on the client. Use commas (,) to separate multiple values of *IP address:Port number*. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT).
kafka.topic	flume-channel	Specifies the Kafka topic used by the channel to cache data.
kafka.consumer.group.id	flume	Specifies the data group ID obtained from Kafka. This parameter cannot be left blank.
parseAsFlumeEvent	true	Specifies whether data is parsed into Flume events.
migrateZookeeperOffsets	true	Specifies whether to search for offsets in ZooKeeper and submit them to Kafka when there is no offset in Kafka.

Parameter	Default Value	Description
kafka.consumer.auto.offset.reset	latest	Specifies where to consume if there is no offset record, which can be set to **earliest**, **latest**, or **none**. **earliest** indicates that the offset is reset to the initial point, **latest** indicates that the offset is set to the latest position, and **none** indicates that an exception is thrown if there is no offset.
kafka.producer.security.protocol	SASL_PLAINTEXT	Specifies the Kafka producer security protocol. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT). **NOTE** If the parameter is not displayed, click **+** in the lower left corner of the dialog box to display all parameters.
kafka.consumer.security.protocol	SASL_PLAINTEXT	Specifies the Kafka consumer security protocol. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT).
pollTimeout	500	Specifies the maximum timeout interval for the consumer to invoke the poll function. Unit: milliseconds

Parameter	Default Value	Description
ignoreLongMessage	false	Specifies whether to discard oversized messages.
messageMaxLength	1000012	Specifies the maximum length of a message written by Flume to Kafka.

Common Sink Configurations

- **HDFS Sink**

  An HDFS sink writes data into HDFS. Common configurations are as follows:

  Table 12-115 Common configurations of an HDFS sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink.
type	hdfs	Specifies the type of the hdfs sink, which must be set to **hdfs**.
hdfs.path	-	Specifies the data storage path in HDFS. The value must start with **hdfs://hacluster/**.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the sink is restarted. Unit: second
hdfs.inUseSuffix	.tmp	Specifies the suffix of the HDFS file to which data is being written.
hdfs.rollInterval	30	Specifies the interval for file rolling, expressed in seconds.
hdfs.rollSize	1024	Specifies the size for file rolling, expressed in bytes.
hdfs.rollCount	10	Specifies the number of events for file rolling. **NOTE** Parameters **rollInterval**, **rollSize**, and **rollCount** can be configured at the same time. The parameter meeting the requirements takes precedence for compression.

Parameter	Default Value	Description
hdfs.idleTimeout	0	Specifies the timeout interval for closing idle files automatically, expressed in seconds.
hdfs.batchSize	1000	Specifies the number of events written into HDFS in batches.
hdfs.kerberosPrincipal	-	Specifies the Kerberos principal of HDFS authentication. This parameter is mandatory in a secure mode, but not required in a common mode.
hdfs.kerberosKeytab	-	Specifies the Kerberos keytab of HDFS authentication. This parameter is not required in a common mode, but in a secure mode, the Flume running user must have the permission to access **keyTab** path in the **jaas.cof** file.
hdfs.fileCloseByEndEvent	true	Specifies whether to close the HDFS file when the last event of the source file is received.
hdfs.batchCallTimeout	-	Specifies the timeout control duration when events are written into HDFS in batches. Unit: milliseconds If this parameter is not specified, the timeout duration is controlled when each event is written into HDFS. When the value of **hdfs.batchSize** is greater than 0, configure this parameter to improve the performance of writing data into HDFS. **NOTE** The value of **hdfs.batchCallTimeout** depends on **hdfs.batchSize**. A greater **hdfs.batchSize** requires a larger **hdfs.batchCallTimeout**. If the value of **hdfs.batchCallTimeout** is too small, writing events to HDFS may fail.
serializer.appendNewline	true	Specifies whether to add a line feed character (**\n**) after an event is written to HDFS. If a line feed character is added, the data volume counters used by the line feed character will not be calculated by HDFS sinks.

Parameter	Default Value	Description
hdfs.filePrefix	over_%{basename}	Specifies the file name prefix after data is written to HDFS.
hdfs.fileSuffix	-	Specifies the file name suffix after data is written to HDFS.
hdfs.inUsePrefix	-	Specifies the prefix of the HDFS file to which data is being written.
hdfs.fileType	DataStream	Specifies the HDFS file format, which can be set to **SequenceFile**, **DataStream**, or **CompressedStream**. **NOTE** If the parameter is set to **SequenceFile** or **DataStream**, output files are not compressed, and the **codeC** parameter cannot be configured. However, if the parameter is set to **CompressedStream**, the output files are compressed, and the **codeC** parameter must be configured together.
hdfs.codeC	-	Specifies the file compression format, which can be set to **gzip**, **bzip2**, **lzo**, **lzop**, or **snappy**.
hdfs.maxOpenFiles	5000	Specifies the maximum number of HDFS files that can be opened. If the number of opened files reaches this value, the earliest opened files are closed.
hdfs.writeFormat	Writable	Specifies the file write format, which can be set to **Writable** or **Text**.
hdfs.callTimeout	10000	Specifies the timeout control duration each time events are written into HDFS, expressed in milliseconds.
hdfs.threadsPoolSize	-	Specifies the number of threads used by each HDFS sink for HDFS I/O operations.
hdfs.rollTimerPoolSize	-	Specifies the number of threads used by each HDFS sink to schedule the scheduled file rolling.
hdfs.round	false	Specifies whether to round off the timestamp value. If this parameter is set to true, all time-based escape sequences (except %t) are affected.

Parameter	Default Value	Description
hdfs.roundUnit	second	Specifies the unit of the timestamp value that has been rounded off, which can be set to **second**, **minute**, or **hour**.
hdfs.useLocalTimeStamp	true	Specifies whether to enable the local timestamp. The recommended parameter value is **true**.
hdfs.closeTries	0	Specifies the maximum attempts for the **hdfs sink** to stop renaming a file. If the parameter is set to the default value **0**, the sink does not stop renaming the file until the file is successfully renamed.
hdfs.retryInterval	180	Specifies the interval of request for closing the HDFS file, expressed in seconds. **NOTE** For each closing request, there are multiple RPCs working on the NameNode back and forth, which may make the NameNode overloaded if the parameter value is too small. Also, when the parameter is set to **0**, the Sink will not attempt to close the file, but opens the file or uses **.tmp** as the file name extension, if the first closing attempt fails.
hdfs.failcount	10	Specifies the number of times that data fails to be written to HDFS. If the number of times that the sink fails to write data to HDFS exceeds the parameter value, an alarm indicating abnormal data transmission is reported.

- **Avro Sink**

  An Avro sink converts events into Avro events and sends them to the monitoring ports of the hosts. Common configurations are as follows:

  Table 12-116 Common configurations of an Avro sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink.
type	-	Specifies the type of the avro sink, which must be set to **avro**.

Parameter	Default Value	Description
hostname	-	Specifies the bound host name or IP address.
port	-	Specifies the bound listening port. Ensure that this port is not occupied.
batch-size	1000	Specifies the number of events sent in a batch.
client.type	DEFAULT	Specifies the client instance type. Set this parameter based on the communication protocol used by the configured model. The options are as follows: • **DEFAULT**: The client instance of the AvroRPC type is returned. • **OTHER**: NULL is returned. • **THRIFT**: The client instance of the Thrift RPC type is returned. • **DEFAULT_LOADBALANCING**: The client instance of the LoadBalancing RPC type is returned. • **DEFAULT_FAILOVER**: The client instance of the Failover RPC type is returned.
ssl	false	Specifies whether to use SSL encryption. If this parameter is set to **true**, the values of **keystore** and **keystore-password** must be specified.

Parameter	Default Value	Description
truststore-type	JKS	Specifies the Java trust store type, which can be set to **JKS** or **PKCS12**. **NOTE** Different passwords are used to protect the key store and private key of **JKS**, while the same password is used to protect the key store and private key of **PKCS12**.
truststore	-	Specifies the Java trust store file.
truststore-password	-	Specifies the Java trust store password.
keystore-type	JKS	Specifies the keystore type set after SSL is enabled.
keystore	-	Specifies the keystore file path set after SSL is enabled. This parameter is mandatory if SSL is enabled.
keystore-password	-	Specifies the keystore password after SSL is enabled. This parameter is mandatory if SSL is enabled.
connect-timeout	20000	Specifies the timeout for the first connection, expressed in milliseconds.
request-timeout	20000	Specifies the maximum timeout for a request after the first request, expressed in milliseconds.

Parameter	Default Value	Description
reset-connection-interval	0	Specifies the interval between a connection failure and a second connection, expressed in seconds. If the parameter is set to **0**, the system continuously attempts to perform a connection.
compression-type	none	Specifies the compression type of the batch data, which can be set to **none** or **deflate**. **none** indicates that data is not compressed, while **deflate** indicates that data is compressed. This parameter value must be the same as that of the AvroSource compression-type.
compression-level	6	Specifies the compression level of batch data, which can be set to **1** to **9**. A larger value indicates a higher compression rate.
exclude-protocols	SSLv3	Specifies the excluded protocols. The entered protocols must be separated by spaces. The default value is **SSLv3**.

- **HBase Sink**

  An HBase sink writes data into HBase. Common configurations are as follows:

  Table 12-117 Common configurations of an HBase sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink.

Parameter	Default Value	Description
type	-	Specifies the type of the HBase sink, which must be set to **hbase**.
table	-	Specifies the HBase table name.
columnFamily	-	Specifies the HBase column family.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the sink is restarted. Unit: second
batchSize	1000	Specifies the number of events written into HBase in batches.
kerberosPrincipal	-	Specifies the Kerberos principal of HBase authentication. This parameter is mandatory in a secure mode, but not required in a common mode.
kerberosKeytab	-	Specifies the Kerberos keytab of HBase authentication. This parameter is not required in a common mode, but in a secure mode, the Flume running user must have the permission to access **keyTab** path in the **jaas.cof** file.
coalesceIncrements	true	Specifies whether to perform multiple operations on the same hbase cell in a same processing batch. Setting this parameter to **true** improves performance.

- **Kafka Sink**

  A Kafka sink writes data into Kafka. Common configurations are as follows:

  Table 12-118 Common configurations of a Kafka sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink.
type	-	Specifies the type of the kafka sink, which must be set to **org.apache.flume.sink.kafka.KafkaSink**.

Parameter	Default Value	Description
kafka.bootstrap.servers	-	Specifies the bootstrap address port list of Kafka. If Kafka has been installed in the cluster and the configuration has been synchronized to the server, you do not need to set this parameter on the server. The default value is the list of all brokers in the Kafka cluster. The client must be configured with this parameter. If there are multiple values, use commas (,) to separate the values. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT).
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the sink is restarted. Unit: second
kafka.producer.acks	1	Successful write is determined by the number of received acknowledgement messages about replicas. The value **0** indicates that no confirm message needs to be received, the value **1** indicates that the system is only waiting for only the acknowledgement information from a leader, and the value **-1** indicates that the system is waiting for the acknowledgement messages of all replicas. If this parameter is set to **-1**, data loss can be avoided in some leader failure scenarios.
kafka.topic	-	Specifies the topic to which data is written. This parameter is mandatory.
flumeBatchSize	1000	Specifies the number of events written into Kafka in batches.
kafka.security.protocol	SASL_PLAINTEXT	Specifies the Kafka security protocol. The parameter value must be set to PLAINTEXT in a common cluster. The rules for matching ports and security protocols must be as follows: port 21007 matches the security mode (SASL_PLAINTEXT), and port 9092 matches the common mode (PLAINTEXT).

Parameter	Default Value	Description
ignoreLongMessage	false	Specifies whether to discard oversized messages.
messageMaxLength	1000012	Specifies the maximum length of a message written by Flume to Kafka.
defaultPartitionId	-	Specifies the Kafka partition ID to which the events of a channel is transferred. The **partitionIdHeader** value overwrites this parameter value. By default, if this parameter is left blank, events will be distributed by the Kafka Producer's partitioner (by a specified key or a partitioner customized by **kafka.partitioner.class**).
partitionIdHeader	-	When you set this parameter, the sink will take the value of the field named using the value of this property from the event header and send the message to the specified partition of the topic. If the value does not have a valid partition, **EventDeliveryException** is thrown. If the header value already exists, this setting overwrites the **defaultPartitionId** parameter.
Other Kafka Producer Properties	-	Specifies other Kafka configurations. This parameter can be set to any production configuration supported by Kafka, and the **.kafka** prefix must be added to the configuration.

- **Thrift Sink**

  A Thrift sink converts events to Thrift events and sends them to the monitoring port of the configured host. Common configurations are as follows:

  Table 12-119 Common configurations of a Thrift sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink.
type	thrift	Specifies the type of the thrift sink, which must be set to **thrift**.

Parameter	Default Value	Description
hostname	-	Specifies the bound host name or IP address.
port	-	Specifies the bound listening port. Ensure that this port is not occupied.
batch-size	1000	Specifies the number of events sent in a batch.
connect-timeout	20000	Specifies the timeout for the first connection, expressed in milliseconds.
request-timeout	20000	Specifies the maximum timeout for a request after the first request, expressed in milliseconds.
kerberos	false	Specifies whether Kerberos authentication is enabled.
client-keytab	-	Specifies the path of the client **keytab** file. The Flume running user must have the access permission on the authentication file.
client-principal	-	Specifies the principal of the security user used by the client.
server-principal	-	Specifies the principal of the security user used by the server.
compression-type	none	Specifies the compression type of data sent by Flume, which can be set to **none** or **deflate**. **none** indicates that data is not compressed, while **deflate** indicates that data is compressed.

Parameter	Default Value	Description
maxConnections	5	Specifies the maximum size of the connection pool for Flume to send data.
ssl	false	Specifies whether to use SSL encryption.
truststore-type	JKS	Specifies the Java trust store type.
truststore	-	Specifies the Java trust store file.
truststore-password	-	Specifies the Java trust store password.
reset-connection-interval	0	Specifies the interval between a connection failure and a second connection, expressed in seconds. If the parameter is set to **0**, the system continuously attempts to perform a connection.

Precautions

- What are the reliability measures of Flume?
  - Use the transaction mechanisms between Source and Channel as well as between Channel and Sink.
  - Configure the failover and load_balance mechanisms for Sink Processor. The following shows a load balancing example. For details, see **http://flume.apache.org/releases/1.9.0.html**.

```
server.sinkgroups=g1
server.sinkgroups.g1.sinks=k1 k2
server.sinkgroups.g1.processor.type=load_balance
server.sinkgroups.g1.processor.backoff=true
server.sinkgroups.g1.processor.selector=random
```

- What are the precautions for the aggregation and cascading of multiple Flume agents?
  - Avro or Thrift protocol can be used for cascading.
  - When the aggregation end contains multiple nodes, evenly distribute the agents and do not aggregate all agents on a single node.

12.7.8 Flume Configuration Parameter Description

For versions earlier than MRS 3.x, configure Flume parameters in the **properties.properties** file.

For MRS 3.x or later, some parameters can be configured on Manager.

Overview

This section describes how to configure the sources, channels, and sinks of Flume, and modify the configuration items of each module.

For MRS 3.x or later, log in to FusionInsight Manager and choose **Cluster** > **Services** > **Flume**. On the displayed page, click the **Configuration Tool** tab, select and drag the source, channel, and sink to be used to the GUI on the right, and double-click them to configure corresponding parameters. Parameters such as **channels** and **type** are configured only in the client configuration file **properties.properties**, the path of which is *Flume client installation directory*/**fusioninsight-flume-***Flume version*/**conf/properties.properties**.

NOTE

You must input encrypted information for some configurations. For details on how to encrypt information, see **Using the Encryption Tool of the Flume Client**.

Common Source Configurations

- **Avro Source**

  An Avro source listens to the Avro port, receives data from the external Avro client, and places data into configured channels. **Table 12-120** lists common configurations.

  Table 12-120 Common configurations of an Avro source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured. Use spaces to separate them. In a single proxy process, sources and sinks are connected through channels. A source instance corresponds to multiple channels, but a sink instance corresponds only to one channel. The format is as follows: **<Agent >.sources.<Source>.channels = <channel1> <channel2> <channel3>...** **<Agent >.sinks.<Sink>.channels = <channel1>** This parameter can be configured only in the **properties.properties** file.

Parameter	Default Value	Description
type	avro	Specifies the type, which is set to **avro**. The type of each source is a fixed value. This parameter can be configured only in the **properties.properties** file.
bind	-	Specifies the host name or IP address associated with the source.
port	-	Specifies the bound port number.
ssl	false	Specifies whether to use SSL encryption. ● true ● false
truststore-type	JKS	Specifies the Java trust store type. Set this parameter to **JKS** or other truststore types supported by Java.
truststore	-	Specifies the Java trust store file.
truststore-password	-	Specifies the Java trust store password.
keystore-type	JKS	Specifies the key storage type. Set this parameter to **JKS** or other truststore types supported by Java.
keystore	-	Specifies the key storage file.
keystore-password	-	Specifies the key storage password.

- **SpoolDir Source**

  A SpoolDir source monitors and transmits new files that have been added to directories in quasi-real-time mode. Common configurations are as follows:

  Table 12-121 Common configurations of a SpoolDir source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured. This parameter can be configured only in the **properties.properties** file.

Parameter	Default Value	Description
type	spooldir	Type, which is set to **spooldir**. This parameter can be configured only in the **properties.properties** file.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the source is restarted. Unit: second
spoolDir	-	Specifies the monitoring directory.
fileSuffix	.COMPLETED	Specifies the suffix added after file transmission is complete.
deletePolicy	never	Specifies the source file deletion policy after file transmission is complete. The value can be either **never** or **immediate**.
ignorePattern	^$	Specifies the regular expression of a file to be ignored.
trackerDir	.flumespool	Specifies the metadata storage path during data transmission.
batchSize	1000	Specifies the source transmission granularity.
decodeErrorPolicy	FAIL	Specifies the code error policy. This parameter can be configured only in the **properties.properties** file. The value can be **FAIL**, **REPLACE**, or **IGNORE**. **FAIL**: Generate an exception and fail the parsing. **REPLACE**: Replace the characters that cannot be identified with other characters, such as U+FFFD. **IGNORE**: Discard character strings that cannot be parsed. **NOTE** If a code error occurs in the file, set **decodeErrorPolicy** to **REPLACE** or **IGNORE**. Flume will skip the code error and continue to collect subsequent logs.

Parameter	Default Value	Description
deserializer	LINE	Specifies the file parser. The value can be either **LINE** or **BufferedLine**. • When the value is set to **LINE**, characters read from the file are transcoded one by one. • When the value is set to **BufferedLine**, one line or multiple lines of characters read from the file are transcoded in batches, which delivers better performance.
deserializer.maxLineLength	2048	Specifies the maximum length for resolution by line, ranging from 0 to 2,147,483,647.
deserializer.maxBatchLine	1	Specifies the maximum number of lines for resolution by line. If multiple lines are set, **maxLineLength** must be set to a corresponding multiplier. For example, if **maxBatchLine** is set to **2**, **maxLineLength** is set to **4096** (2048 x 2).
selector.type	replicating	Specifies the selector type. The value can be either **replicating** or **multiplexing**. • **replicating** indicates that the same content is sent to each channel. • **multiplexing** indicates that the content is sent only to certain channels according to the distribution rule.
interceptors	-	Specifies the interceptor. For details, see the **Flume official document**. This parameter can be configured only in the **properties.properties** file.

NOTE

The Spooling source ignores the last line feed character of each event when data is read by line. Therefore, Flume does not calculate the data volume counters used by the last line feed character.

- **Kafka Source**

  A Kafka source consumes data from Kafka topics. Multiple sources can consume data of the same topic, and the sources consume different partitions of the topic. Common configurations are as follows:

  Table 12-122 Common configurations of a Kafka source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured. This parameter can be configured only in the **properties.properties** file.
type	org.apache.flume.source.kafka.KafkaSource	Specifies the type, which is set to **org.apache.flume.source.kafka.KafkaSource**. This parameter can be configured only in the **properties.properties** file.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the source is restarted. Unit: second
nodatatime	0 (Disabled)	Specifies the alarm threshold. An alarm is triggered when the duration that Kafka does not release data to subscribers exceeds the threshold. Unit: second
batchSize	1000	Specifies the number of events written into a channel at a time.
batchDurationMillis	1000	Specifies the maximum duration of topic data consumption at a time, expressed in milliseconds.
keepTopicInHeader	false	Specifies whether to save topics in the event header. If topics are saved, topics configured in Kafka sinks become invalid. ● true ● false This parameter can be configured only in the **properties.properties** file.

Parameter	Default Value	Description
keepPartitionIn-Header	false	Specifies whether to save partition IDs in the event header. If partition IDs are saved, Kafka sinks write data to the corresponding partitions. ● true ● false This parameter can be set only in the properties.properties file.
kafka.bootstrap.servers	-	Specifies the list of Broker addresses, which are separated by commas.
kafka.consumer.group.id	-	Specifies the Kafka consumer group ID.
kafka.topics	-	Specifies the list of subscribed Kafka topics, which are separated by commas (,).
kafka.topics.regex	-	Specifies the subscribed topics that comply with regular expressions. **kafka.topics.regex** has a higher priority than **kafka.topics** and will overwrite **kafka.topics**.
kafka.security.protocol	SASL_PLAINTEXT	Specifies the security protocol of Kafka. The value must be set to **PLAINTEXT** for clusters in which Kerberos authentication is disabled.
kafka.kerberos.domain.name	-	Specifies the value of **default_realm** of Kerberos in the Kafka cluster, which should be configured only for security clusters. This parameter can be set only in the properties.properties file.
Other Kafka Consumer Properties	-	Specifies other Kafka configurations. This parameter can be set to any consumption configuration supported by Kafka, and the **.kafka** prefix must be added to the configuration. This parameter can be set only in the properties.properties file.

- **Taildir Source**

A Taildir source monitors file changes in a directory and automatically reads the file content. In addition, it can transmit data in real time. **Table 12-123** lists common configurations.

Table 12-123 Common configurations of a Taildir source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured. This parameter can be set only in the properties.properties file.
type	taildir	Specifies the type, which is set to **taildir**. This parameter can be set only in the properties.properties file.
filegroups	-	Specifies the group name of a collection file directory. Group names are separated by spaces.
filegroups.<filegroupName>.parentDir	-	Specifies the parent directory. The value must be an absolute path. This parameter can be set only in the properties.properties file.
filegroups.<filegroupName>.filePattern	-	Specifies the relative file path of the file group's parent directory. Directories can be included and regular expressions are supported. It must be used together with **parentDir**. This parameter can be set only in the properties.properties file.
positionFile	-	Specifies the metadata storage path during data transmission.
headers.<filegroupName>.<headerKey>	-	Specifies the key-value of an event when data of a group is being collected. This parameter can be set only in the properties.properties file.
byteOffsetHeader	false	Specifies whether each event header should contain the location information about the event in the source file. The location information is saved in the **byteoffset** variable.

Parameter	Default Value	Description
skipToEnd	false	Specifies whether Flume can locate the latest location of a file and read the latest data after restart.
idleTimeout	120000	Specifies the idle duration during file reading, expressed in milliseconds. If the file data is not changed in this idle period, the source closes the file. If data is written into this file after it is closed, the source opens the file and reads data.
writePosInterval	3000	Specifies the interval for writing metadata to a file, expressed in milliseconds.
batchSize	1000	Specifies the number of events written to the channel in batches.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the source is restarted. Unit: second

- **Http Source**

  An HTTP source receives data from an external HTTP client and sends the data to the configured channels. **Table 12-124** lists common configurations.

  Table 12-124 Common configurations of an HTTP source

Parameter	Default Value	Description
channels	-	Specifies the channel connected to the source. Multiple channels can be configured. This parameter can be set only in the properties.properties file.
type	http	Specifies the type, which is set to **http**. This parameter can be set only in the properties.properties file.
bind	-	Specifies the name or IP address of the bound host.
port	-	Specifies the bound port.

Parameter	Default Value	Description
handler	org.apache.flume.source.http.JSONHandler	Specifies the message parsing method of an HTTP request. The following methods are supported: • **org.apache.flume.source.http.JSONHandler**: JSON • **org.apache.flume.sink.solr.morphline.BlobHandler**: BLOB
handler.*	-	Specifies handler parameters.
enableSSL	false	Specifies whether SSL is enabled in HTTP.
keystore	-	Specifies the keystore path set after SSL is enabled in HTTP.
keystorePassword	-	Specifies the keystore password set after SSL is enabled in HTTP.

Common Channel Configurations

- **Memory Channel**

  A memory channel uses memory as the cache. Events are stored in memory queues. **Table 12-125** lists common configurations.

  Table 12-125 Common configurations of a memory channel

Parameter	Default Value	Description
type	-	Specifies the type, which is set to **memory**. This parameter can be set only in the properties.properties file.
capacity	10000	Specifies the maximum number of events cached in a channel.
transactionCapacity	1000	Specifies the maximum number of events accessed each time.
channelfullcount	10	Specifies the channel full count. When the count reaches the threshold, an alarm is reported.

- **File Channel**

  A file channel uses local disks as the cache. Events are stored in the folder specified by **dataDirs**. **Table 12-126** lists common configurations.

Table 12-126 Common configurations of a file channel

Parameter	Default Value	Description
type	-	Specifies the type, which is set to **file**. This parameter can be set only in the properties.properties file.
checkpointDir	${BIGDATA_DATA_HOME}/flume/checkpoint	Specifies the checkpoint storage directory.
dataDirs	${BIGDATA_DATA_HOME}/flume/data	Specifies the data cache directory. Multiple directories can be configured to improve performance. The directories are separated by commas (,).
maxFileSize	2146435071	Specifies the maximum size of a single cache file, expressed in bytes.
minimumRequired-Space	524288000	Specifies the minimum idle space in the cache, expressed in bytes.
capacity	1000000	Specifies the maximum number of events cached in a channel.
transactionCapacity	10000	Specifies the maximum number of events accessed each time.
channelfullcount	10	Specifies the channel full count. When the count reaches the threshold, an alarm is reported.

- **Kafka Channel**

  A Kafka channel uses a Kafka cluster as the cache. Kafka provides high availability and multiple copies to prevent data from being immediately consumed by sinks when Flume or Kafka Broker crashes. **Table 10 Common configurations of a Kafka channel** lists common configurations.

  Table 12-127 Common configurations of a Kafka channel

Parameter	Default Value	Description
type	-	Specifies the type, which is set to **org.apache.flume.channel.kafka.KafkaChannel**. This parameter can be set only in the properties.properties file.

Parameter	Default Value	Description
kafka.bootstrap.servers	-	Specifies the list of Brokers in the Kafka cluster.
kafka.topic	flume-channel	Specifies the Kafka topic used by the channel to cache data.
kafka.consumer.group.id	flume	Specifies the Kafka consumer group ID.
parseAsFlumeEvent	true	Specifies whether data is parsed into Flume events.
migrateZookeeper-Offsets	true	Specifies whether to search for offsets in ZooKeeper and submit them to Kafka when there is no offset in Kafka.
kafka.consumer.auto.offset.reset	latest	Consumes data from the specified location when there is no offset.
kafka.producer.security.protocol	SASL_PLAINTEXT	Specifies the Kafka producer security protocol.
kafka.consumer.security.protocol	SASL_PLAINTEXT	Specifies the Kafka consumer security protocol.

Common Sink Configurations

- **HDFS Sink**

  An HDFS sink writes data into HDFS. **Table 12-128** lists common configurations.

  Table 12-128 Common configurations of an HDFS sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink. This parameter can be set only in the properties.properties file.
type	hdfs	Specifies the type, which is set to **hdfs**. This parameter can be set only in the properties.properties file.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the sink is restarted. Unit: second
hdfs.path	-	Specifies the HDFS path.

Parameter	Default Value	Description
hdfs.inUseSuffix	.tmp	Specifies the suffix of the HDFS file to which data is being written.
hdfs.rollInterval	30	Specifies the interval for file rolling, expressed in seconds.
hdfs.rollSize	1024	Specifies the size for file rolling, expressed in bytes.
hdfs.rollCount	10	Specifies the number of events for file rolling.
hdfs.idleTimeout	0	Specifies the timeout interval for closing idle files automatically, expressed in seconds.
hdfs.batchSize	1000	Specifies the number of events written into HDFS at a time.
hdfs.kerberosPrincipal	-	Specifies the Kerberos username for HDFS authentication. This parameter is not required for a cluster in which Kerberos authentication is disabled.
hdfs.kerberosKeytab	-	Specifies the Kerberos keytab of HDFS authentication. This parameter is not required for a cluster in which Kerberos authentication is disabled.
hdfs.fileCloseByEndEvent	true	Specifies whether to close the file when the last event is received.
hdfs.batchCallTimeout	-	Specifies the timeout control duration each time events are written into HDFS, expressed in milliseconds. If this parameter is not specified, the timeout duration is controlled when each event is written into HDFS. When the value of **hdfs.batchSize** is greater than 0, configure this parameter to improve the performance of writing data into HDFS. **NOTE** The value of **hdfs.batchCallTimeout** depends on **hdfs.batchSize**. A greater **hdfs.batchSize** requires a larger **hdfs.batchCallTimeout**. If the value of **hdfs.batchCallTimeout** is too small, writing events to HDFS may fail.

Parameter	Default Value	Description
serializer.appendNewline	true	Specifies whether to add a line feed character (**\n**) after an event is written to HDFS. If a line feed character is added, the data volume counters used by the line feed character will not be calculated by HDFS sinks.

- **Avro Sink**

  An Avro sink converts events into Avro events and sends them to the monitoring ports of the hosts. **Table 12-129** lists common configurations.

  Table 12-129 Common configurations of an Avro sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink. This parameter can be set only in the properties.properties file.
type	-	Specifies the type, which is set to **avro**. This parameter can be set only in the properties.properties file.
hostname	-	Specifies the name or IP address of the bound host.
port	-	Specifies the monitoring port.
batch-size	1000	Specifies the number of events sent in a batch.
ssl	false	Specifies whether to use SSL encryption.
truststore-type	JKS	Specifies the Java trust store type.
truststore	-	Specifies the Java trust store file.
truststore-password	-	Specifies the Java trust store password.
keystore-type	JKS	Specifies the key storage type.
keystore	-	Specifies the key storage file.
keystore-password	-	Specifies the key storage password.

- **HBase Sink**

  An HBase sink writes data into HBase. **Table 12-130** lists common configurations.

Table 12-130 Common configurations of an HBase sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink. This parameter can be set only in the properties.properties file.
type	-	Specifies the type, which is set to **hbase**. This parameter can be set only in the properties.properties file.
table	-	Specifies the HBase table name.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the sink is restarted. Unit: second
columnFamily	-	Specifies the HBase column family.
batchSize	1000	Specifies the number of events written into HBase at a time.
kerberosPrincipal	-	Specifies the Kerberos username for HBase authentication. This parameter is not required for a cluster in which Kerberos authentication is disabled.
kerberosKeytab	-	Specifies the Kerberos keytab of HBase authentication. This parameter is not required for a cluster in which Kerberos authentication is disabled.

- **Kafka Sink**

  A Kafka sink writes data into Kafka. **Table 12-131** lists common configurations.

  Table 12-131 Common configurations of a Kafka sink

Parameter	Default Value	Description
channel	-	Specifies the channel connected to the sink. This parameter can be set only in the properties.properties file.

Parameter	Default Value	Description
type	-	Specifies the type, which is set to **org.apache.flume.sink.kafka.KafkaSink**. This parameter can be set only in the properties.properties file.
kafka.bootstrap.servers	-	Specifies the list of Kafka Brokers, which are separated by commas.
monTime	0 (Disabled)	Specifies the thread monitoring threshold. When the update time exceeds the threshold, the sink is restarted. Unit: second
kafka.topic	default-flume-topic	Specifies the topic where data is written.
flumeBatchSize	1000	Specifies the number of events written into Kafka at a time.
kafka.security.protocol	SASL_PLAINTEXT	Specifies the security protocol of Kafka. The value must be set to **PLAINTEXT** for clusters in which Kerberos authentication is disabled.
kafka.kerberos.domain.name	-	Specifies the Kafka domain name. This parameter is mandatory for a security cluster. This parameter can be set only in the properties.properties file.
Other Kafka Producer Properties	-	Specifies other Kafka configurations. This parameter can be set to any production configuration supported by Kafka, and the **.kafka** prefix must be added to the configuration. This parameter can be set only in the properties.properties file.

12.7.9 Using Environment Variables in the properties.properties File

Scenario

This section describes how to use environment variables in the **properties.properties** configuration file.

This section applies to MRS 3.*x* and later versions.

Prerequisites

You have installed Flume service or client.

Procedure

Step 1 Add variables to the *Flume client installation directory*/fusioninsight-flume-*Flume Version*/**conf/flume-env.sh** file.

Add variables:

export *Variable name* **=** *Variable value*

Example:

JAVA_OPTS="-Xms2G -Xmx4G -XX:CMSFullGCsBeforeCompaction=1 -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:+UseCMSCompactAtFullCollection -DpropertiesImplementation=org.apache.flume.node.EnvVarResolverProperties "

export TAILDIR_PATH=/tmp/flumetest/201907/20190703/1/.*log.*

Step 2 Restart the Flume instance process.

1. Log in to FusionInsight Manager.
2. Choose **Cluster** > *Name of the target cluster* > **Services** > **Flume** > **Instance**, select all Flume instances, choose **More** > **Restart Instance**, enter the password, and click **OK**.

NOTICE

- Do not restart the Flume process by restarting the Flume service on the Manager page, after **flume-env.sh** takes effect in the flume server. Otherwise, user-defined environment variables are lost.
- Ensure that **flume-env.sh** takes effect before configuring the **properties.properties** as instructed in **Step 3** and uploading the file on the GUI. If the operation sequence is not standard, user-defined environment variables may be lost.

Step 3 In the **properties.properties** configuration file, use the **$**{variable name} format to reference the variable, taking the client as an example.

Example:

```
client.sources.s1.type = TAILDIR
client.sources.s1.filegroups = f1
client.sources.s1.filegroups.f1 = ${TAILDIR_PATH}
client.sources.s1.positionFile = /tmp/flumetest/201907/20190703/1/taildir_position.json
client.sources.s1.channels = c1
```

----End

12.7.10 Non-Encrypted Transmission

12.7.10.1 Configuring Non-encrypted Transmission

Scenario

This section describes how to configure Flume server and client parameters after the cluster and the Flume service are installed to ensure proper running of the service.

This section applies to MRS 3.*x* and later versions.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**.

Prerequisites

- The cluster and Flume service have been installed.
- The network environment of the cluster is secure.

Procedure

Step 1 Configure the client parameters of the Flume role.

1. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select and drag the source, channel, and sink to be used to the GUI on the right, and connect them.

      For example, use SpoolDir Source, File Channel, and Avro Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by referring to **Table 12-132** based on the actual environment.

      NOTE

      - If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > **Services** > **Flume** > **Configuration** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
   d. Click **Export** to save the **properties.properties** configuration file to the local server.

Table 12-132 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the function is not enabled.	false

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 2 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.
   a. Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **server**. Select and drag the source, channel, and sink to be used to the GUI on the right, and connect them.
      For example, use Avro Source, File Channel, and HDFS Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by referring to **Table 12-133** based on the actual environment.

      NOTE

      ■ If the server parameters of the Flume role have been configured, you can choose **Cluster** > **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > **Service** > **Flume** > **Configurations** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.

      ■ It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

      ■ A unique checkpoint directory needs to be configured for each File Channel.
   d. Click **Export** to save the **properties.properties** configuration file to the local server.

Table 12-133 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the function is not enabled.	false

2. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Flume**. On the **Instances** tab page, click **Flume**.
3. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
4. Click **Save**, and then click **OK**.
5. Click **Finish**.

----End

12.7.10.2 Typical Scenario: Collecting Local Static Logs and Uploading Them to Kafka

Scenario

This section describes how to use Flume to collect static logs from a local host (service IP address: 192.168.108.11) and save them to the topic list (test1) of Kafka.

This section applies to MRS 3.*x* and later versions.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**. The configuration can apply to scenarios where only the server is configured, for example, Server:Spooldir Source+File Channel+Kafka Sink.

Prerequisites

- The cluster, Kafka, and Flume service have been installed.
- The network environment of the cluster is secure.
- The system administrator has understood service requirements and prepared Kafka administrator **flume_kafka**.

Procedure

Step 1 Configure the client parameters of the Flume role.

1. Use the Flume configuration tool on Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select and drag the source, channel, and sink to be used to the GUI on the right, and connect them.

      Use SpoolDir Source, File Channel, and Avro Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by referring to **Table 12-134** based on the actual environment.

      NOTE

      - If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to Manager, choose **Cluster** > **Services** > **Flume** > **Configurations** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
   d. Click **Export** to save the **properties.properties** configuration file to the local server.

Table 12-134 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test

Parameter	Description	Example Value
spoolDir	Specifies the directory where the file to be collected resides. This parameter cannot be left blank. The directory needs to exist and have the write, read, and execute permissions on the flume running user.	/srv/BigData/hadoop/data1/zb
trackerDir	Specifies the path for storing the metadata of files collected by Flume.	/srv/BigData/hadoop/data1/tracker
batchSize	Specifies the number of events that Flume sends in a batch (number of data pieces). A larger value indicates higher performance and lower timeliness.	61200
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data

Parameter	Description	Example Value
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hostname	Specifies the name or IP address of the host whose data is to be sent. This parameter cannot be left blank. The parameter must be configured to be the name or IP address of the host where the connected Avro Source resides.	192.168.108.11

Parameter	Description	Example Value
port	Specifies the port that sends the data. This parameter cannot be left blank. It must be configured to be the port that is listened to by the connected Avro Source.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the function is not enabled.	false

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 2 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.
   a. Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **server**. Select and drag the source, channel, and sink to be used to the GUI on the right, and connect them.
      Avro Source, File Channel, and Kafka Sink are used.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by referring to **Table 12-135** based on the actual environment.

NOTE

- If the server parameters of the Flume role have been configured, you can choose **Cluster** > **Services** > **Flume** > **Instance** on Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > **Services** > **Flume** > **Configurations** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
- A unique checkpoint directory needs to be configured for each File Channel.

d. Click **Export** to save the **properties.properties** configuration file to the local server.

Table 12-135 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
bind	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as the IP address that the server configuration file will upload.	192.168.108.11
port	Specifies the port that Avro Source listens to. This parameter cannot be left blank. It must be configured as an unused port.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the function is not enabled.	false

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
kafka.topics	Specifies the list of subscribed Kafka topics, which are separated by commas (,). This parameter cannot be left blank.	test1

Parameter	Description	Example Value
kafka.bootstrap.servers	Specifies the bootstrap IP address and port list of Kafka. The default value is all Kafka lists in a Kafka cluster. If Kafka has been installed in the cluster and its configurations have been synchronized, this parameter can be left blank.	192.168.101.10:21007

2. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Flume**. On the **Instance** tab page, click the **Flume** role.
3. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
4. Click **Save**, and then click **OK**.
5. Click **Finish**.

Step 3 Verify log transmission.

1. Log in to the Kafka client.

   cd /*Client installation directory***/Kafka/kafka**

   kinit flume_kafka (Enter the password.)
2. Read data from a Kafka topic.

   bin/kafka-console-consumer.sh --topic *topic name* **--bootstrap-server** *Kafka service IP address of the node where the role instance is located*:**21007 --consumer.config config/consumer.properties --from-beginning**

   The system displays the contents of the file to be collected.

   ```
   [root@host1 kafka]# bin/kafka-console-consumer.sh --topic test1 --bootstrap-server 192.168.101.10:21007 --consumer.config config/consumer.properties --from-beginning
   Welcome to flume
   ```

----End

12.7.10.3 Typical Scenario: Collecting Local Static Logs and Uploading Them to HDFS

Scenario

This section describes how to use Flume to collect static logs from a local PC (for example, the service IP address is 192.168.108.11) and save them to the **/flume/test** directory on HDFS.

This section applies to MRS 3.*x* or later.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**. This configuration can use only one Flume scenario, for example, Server:Spooldir Source+File Channel+HDFS Sink.

Prerequisites

- The cluster, HDFS, and Flume service have been installed.
- The network environment of the cluster is secure.
- User **flume_hdfs** has been created, and the HDFS directory and data used for log verification have been authorized to the user.

Procedure

Step 1 On FusionInsight Manager, choose **System** > **Permission > User**, select user **flume_hdfs**, and choose **More** > **Download Authentication Credential** to download the Kerberos certificate file of user **flume_hdfs** and save it to the local host.

Step 2 Configure the client parameters of the Flume role.

1. Use Flume on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      Use SpoolDir Source, File Channel, and Avro Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by referring to **Table 12-136** based on the actual environment.

      NOTE

      - If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
   d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-136 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
spoolDir	Specifies the directory where the file to be collected resides. This parameter cannot be left blank. The directory needs to exist and have the write, read, and execute permissions on the flume running user.	/srv/BigData/hadoop/data1/zb
trackerDir	Specifies the path for storing the metadata of files collected by Flume.	/srv/BigData/hadoop/data1/tracker
batch-size	Specifies the number of events that Flume sends in a batch.	61200
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data

Parameter	Description	Example Value
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hostname	Specifies the name or IP address of the host whose data is to be sent. This parameter cannot be left blank. Name or IP address must be configured to be the name or IP address that the Avro source associated with it.	192.168.108.11

Parameter	Description	Example Value
port	Specifies the IP address to which Avro Sink is bound. This parameter cannot be left blank. It must be consistent with the port that is monitored by the connected Avro Source.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 3 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.
   a. Log in to FusionInsight Manager. Choose **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **server**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.
      For example, use Avro Source, File Channel, and HDFS Sink.
   c. Double-click the source, channel, and sink. Refer to **Table 12-137** to set corresponding configuration parameters based on the actual environment.

NOTE

- If the server parameters of the Flume role have been configured, you can choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
- A unique checkpoint directory needs to be configured for each File Channel.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-137 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
bind	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as the IP address that the server configuration file will upload.	192.168.108.11
port	Specifies the ID of the port that the Avro Source monitors. This parameter cannot be left blank. It must be configured as an unused port.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hdfs.path	Specifies the HDFS data write directory. This parameter cannot be left blank.	hdfs://hacluster/flume/test
hdfs.inUsePrefix	Specifies the prefix of the file that is being written to HDFS.	TMP_
hdfs.batchSize	Specifies the maximum number of events that can be written to HDFS once.	61200

Parameter	Description	Example Value
hdfs.kerberosPrincipal	Specifies the Kerberos authentication user, which is mandatory in security versions. This configuration is required only in security clusters.	flume_hdfs
hdfs.kerberosKeytab	Specifies the keytab file path for Kerberos authentication, which is mandatory in security versions. This configuration is required only in security clusters.	/opt/test/conf/user.keytab **NOTE** Obtain the **user.keytab** file from the Kerberos certificate file of the user **flume_hdfs**. In addition, ensure that the user who installs and runs the Flume client has the read and write permissions on the **user.keytab** file.
hdfs.useLocalTimeStamp	Specifies whether to use the local time. Possible values are **true** and **false**.	true

2. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume**. On the displayed page, click the **Flume** role under **Role**.
3. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
4. Click **Save**, and then click **OK**.
5. Click **Finish**.

Step 4 Verify log transmission.

1. Log in to FusionInsight Manager as a user who has the management permission on HDFS. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster >** *Name of the desired cluster* > **Services** > **HDFS**, click the HDFS WebUI link next to **NameNode (Active)** to go to the HDFS WebUI, and choose **Utilities** > **Browse the file system**.
2. Check whether the data is generated in the **/flume/test** directory on the HDFS.

----End

12.7.10.4 Typical Scenario: Collecting Local Dynamic Logs and Uploading Them to HDFS

Scenario

This section describes how to use Flume to collect dynamic logs from a local PC (for example, the service IP address is 192.168.108.11) and save them to the **/flume/test** directory on HDFS.

This section applies to MRS 3.*x* or later.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**. This configuration can use only one Flume scenario, for example, Server:Taildir Source+File Channel+HDFS Sink.

Prerequisites

- The cluster, HDFS, and Flume service have been installed.
- The network environment of the cluster is secure.
- You have created user **flume_hdfs** and authorized the HDFS directory and data to be operated during log verification.

Procedure

Step 1 On FusionInsight Manager, choose **System > User** and choose **More > Download Authentication Credential** to download the Kerberos certificate file of user **flume_hdfs** and save it to the local host.

Step 2 Configure the client parameters of the Flume role.

1. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      For example, use Taildir Source, File Channel, and Avro Sink.
   c. Double-click the source, channel, and sink. Refer to **Table 12-138** to set corresponding configuration parameters based on the actual environment.

NOTE

- If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-138 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
filegroups	Specifies the file group list name. This parameter cannot be left blank. Values are separated by spaces	epgtest
positionFile	Specifies the location where the collected file information (file name and location from which the file collected) is saved. This parameter cannot be left blank. The file does not need to be created manually, but the Flume running user needs to have the write permission on its upper-level directory.	/home/omm/flume/positionfile
batch-size	Specifies the number of events that Flume sends in a batch.	61200

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint

Parameter	Description	Example Value
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hostname	Specifies the name or IP address of the host whose data is to be sent. This parameter cannot be left blank. Name or IP address must be configured to be the name or IP address that the Avro source associated with it.	192.168.108.11
port	Specifies the IP address to which Avro Sink is bound. This parameter cannot be left blank. It must be consistent with the port that is monitored by the connected Avro Source.	21154

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 3 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **server**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.
      For example, use Avro Source, File Channel, and HDFS Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-139** based on the actual environment.

NOTE

- If the server parameters of the Flume role have been configured, you can choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
- A unique checkpoint directory needs to be configured for each File Channel.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-139 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
bind	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as the IP address that the server configuration file will upload.	192.168.108.11
port	Specifies the ID of the port that the Avro Source monitors. This parameter cannot be left blank. It must be configured as an unused port.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hdfs.path	Specifies the HDFS data write directory. This parameter cannot be left blank.	hdfs://hacluster/flume/test
hdfs.inUsePrefix	Specifies the prefix of the file that is being written to HDFS.	TMP_
hdfs.batchSize	Specifies the maximum number of events that can be written to HDFS once.	61200

Parameter	Description	Example Value
hdfs.kerberosPrincipal	Specifies the Kerberos authentication user, which is mandatory in security versions. This configuration is required only in security clusters.	flume_hdfs
hdfs.kerberosKeytab	Specifies the keytab file path for Kerberos authentication, which is mandatory in security versions. This configuration is required only in security clusters.	/opt/test/conf/user.keytab **NOTE** Obtain the **user.keytab** file from the Kerberos certificate file of the user **flume_hdfs**. In addition, ensure that the user who installs and runs the Flume client has the read and write permissions on the **user.keytab** file.
hdfs.useLocalTimeStamp	Specifies whether to use the local time. Possible values are **true** and **false**.	true

2. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume**. On the displayed page, click the **Flume** role in the **Role** column.
3. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
4. Click **Save**, and then click **OK**.
5. Click **Finish**.

Step 4 Verify log transmission.

1. Log in to FusionInsight Manager as a user who has the management permission on HDFS. For details, see**Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster >** *Name of the desired cluster* > **Services** > **HDFS**, click the HDFS web UI link of **NameNode**(*Node name*, **Active**) to go to the HDFS web UI, and choose **Utilities** > **Browse the file system**.
2. Check whether the data is generated in the **/flume/test** directory on the HDFS.

----End

12.7.10.5 Typical Scenario: Collecting Logs from Kafka and Uploading Them to HDFS

Scenario

This section describes how to use Flume to collect logs from the Topic list (test1) of Kafka and save them to the **/flume/test** directory on HDFS.

This section applies to MRS 3.*x* or later.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**. This configuration can use only one Flume scenario, for example, Server:Kafka Source+File Channel+HDFS Sink.

Prerequisites

- The cluster, HDFS, Kafka, and Flume service have been installed.
- The network environment of the cluster is secure.
- You have created user **flume_hdfs** and authorized the HDFS directory and data to be operated during log verification.

Procedure

Step 1 On FusionInsight Manager, choose **System > User** and choose **More > Download Authentication Credential** to download the Kerberos certificate file of user **flume_hdfs** and save it to the local host.

Step 2 Configure the client parameters of the Flume role.

1. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      For example, use Kafka Source, File Channel, and Avro Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-140** based on the actual environment.

NOTE

- If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-140 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
kafka.topics	Specifies the subscribed Kafka topic list, in which topics are separated by commas (,). This parameter cannot be left blank.	test1
kafka.consumer.group.id	Specifies the data group ID obtained from Kafka. This parameter cannot be left blank.	flume
kafka.bootstrap.servers	Specifies the bootstrap IP address and port list of Kafka. The default value is all Kafka lists in a Kafka cluster. If Kafka has been installed in the cluster and its configurations have been synchronized, this parameter can be left blank.	192.168.101.10:9092

Parameter	Description	Example Value
batchSize	Specifies the number of events that Flume sends in a batch (number of data pieces).	61200
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint

Parameter	Description	Example Value
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hostname	Specifies the name or IP address of the host whose data is to be sent. This parameter cannot be left blank. Name or IP address must be configured to be the name or IP address that the Avro source associated with it.	192.168.108.11
port	Specifies the IP address to which Avro Sink is bound. This parameter cannot be left blank. It must be consistent with the port that is monitored by the connected Avro Source.	21154

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 3 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **server**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.
      For example, use Avro Source, File Channel, and HDFS Sink.
   c. Double-click the source, channel, and sink. Refer to **Table 12-141** to set corresponding configuration parameters based on the actual environment.

NOTE

- If the server parameters of the Flume role have been configured, you can choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configurations** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
- A unique checkpoint directory needs to be configured for each File Channel.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-141 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
bind	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as the IP address that the server configuration file will upload.	192.168.108.11
port	Specifies the ID of the port that the Avro Source monitors. This parameter cannot be left blank. It must be configured as an unused port.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hdfs.path	Specifies the HDFS data write directory. This parameter cannot be left blank.	hdfs://hacluster/flume/test
hdfs.inUsePrefix	Specifies the prefix of the file that is being written to HDFS.	TMP_
hdfs.batchSize	Specifies the maximum number of events that can be written to HDFS once.	61200

Parameter	Description	Example Value
hdfs.kerberosPrincipal	Specifies the Kerberos authentication user, which is mandatory in security versions. This configuration is required only in security clusters.	flume_hdfs
hdfs.kerberosKeytab	Specifies the keytab file path for Kerberos authentication, which is mandatory in security versions. This configuration is required only in security clusters.	/opt/test/conf/user.keytab **NOTE** Obtain the **user.keytab** file from the Kerberos certificate file of the user **flume_hdfs**. In addition, ensure that the user who installs and runs the Flume client has the read and write permissions on the **user.keytab** file.
hdfs.useLocalTimeStamp	Specifies whether to use the local time. Possible values are **true** and **false**.	true

2. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume**. On the displayed page, click the **Flume** role under **Role**.
3. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
4. Click **Save**, and then click **OK**.
5. Click **Finish**.

Step 4 Verify log transmission.

1. Log in to FusionInsight Manager as a user who has the management permission on HDFS. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster >** *Name of the desired cluster* > **Services** > **HDFS**, click the HDFS web UI link of **NameNode**(*Node name*, **Active**) to go to the HDFS web UI, and choose **Utilities** > **Browse the file system**.
2. Check whether the data is generated in the **/flume/test** directory on the HDFS.

----End

12.7.10.6 Typical Scenario: Collecting Logs from Kafka and Uploading Them to HDFS Through the Flume Client

Scenario

This section describes how to use Flume to collect logs from the Topic list (test1) of Kafka client and save them to the **/flume/test** directory on HDFS.

This section applies to MRS 3.*x* or later.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**.

Prerequisites

- The cluster, HDFS, Kafka, and Flume service have been installed.
- You have created user **flume_hdfs** and authorized the HDFS directory and data to be operated during log verification.
- The network environment of the cluster is secure.

Procedure

Step 1 On FusionInsight Manager, choose **System > User** and choose **More > Download Authentication Credential** to download the Kerberos certificate file of user **flume_hdfs** and save it to the local host.

Step 2 Configure the client parameters of the Flume role.

1. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      For example, use Kafka Source, File Channel, and HDFS Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-142** based on the actual environment.

      NOTE

      - If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-142 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
kafka.topics	Specifies the subscribed Kafka topic list, in which topics are separated by commas (,). This parameter cannot be left blank.	test1
kafka.consumer.group.id	Specifies the data group ID obtained from Kafka. This parameter cannot be left blank.	flume
kafka.bootstrap.servers	Specifies the bootstrap IP address and port list of Kafka. The default value is all Kafka lists in a Kafka cluster. If Kafka has been installed in the cluster and its configurations have been synchronized, this parameter can be left blank.	192.168.101.10:21007
batchSize	Specifies the number of events that Flume sends in a batch (number of data pieces).	61200

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint

Parameter	Description	Example Value
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hdfs.path	Specifies the HDFS data write directory. This parameter cannot be left blank.	hdfs://hacluster/flume/test
hdfs.inUsePrefix	Specifies the prefix of the file that is being written to HDFS.	TMP_
hdfs.batchSize	Specifies the maximum number of events that can be written to HDFS once.	61200
hdfs.kerberosPrincipal	Specifies the Kerberos authentication user, which is mandatory in security versions. This configuration is required only in security clusters.	flume_hdfs
hdfs.kerberosKeytab	Specifies the keytab file path for Kerberos authentication, which is mandatory in security versions. This configuration is required only in security clusters.	/opt/test/conf/user.keytab **NOTE** Obtain the **user.keytab** file from the Kerberos certificate file of the user **flume_hdfs**. In addition, ensure that the user who installs and runs the Flume client has the read and write permissions on the **user.keytab** file.
hdfs.useLocalTimeStamp	Specifies whether to use the local time. Possible values are **true** and **false**.	true

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.
3. To connect the Flume client to the HDFS, you need to add the following configuration:
   a. Download the Kerberos certificate of account **flume_hdfs** and obtain the **krb5.conf** configuration file. Upload the configuration file to the **fusioninsight-flume-1.9.0/conf/** directory on the node where the client is installed.
   b. In **fusioninsight-flume-1.9.0/conf/**, create the **jaas.conf** configuration file.

      vi jaas.conf

      ```
      KafkaClient {
      com.sun.security.auth.module.Krb5LoginModule required
      useKeyTab=true
      keyTab="/opt/test/conf/user.keytab"
      principal="flume_hdfs@<System domain name>"
      useTicketCache=false
      storeKey=true
      debug=true;
      };
      ```

      Values of **keyTab** and **principal** vary depending on the actual situation.
   c. Obtain configuration files **core-site.xml** and **hdfs-site.xml** from **/opt/FusionInsight_Cluster_*<Cluster ID>*_Flume_ClientConfig/Flume/config** and upload them to **fusioninsight-flume-1.9.0/conf/**.
4. Restart the Flume service.

Step 3 Verify log transmission.

1. Log in to FusionInsight Manager as a user who has the management permission on HDFS. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster >** *Name of the desired cluster* > **Services** > **HDFS**, click the HDFS WebUI link of **NameNode** (*Node name*, **Active**) to go to the HDFS WebUI, and choose **Utilities** > **Browse the file system**.
2. Check whether the data is generated in the **/flume/test** directory on the HDFS.

----End

12.7.10.7 Typical Scenario: Collecting Local Static Logs and Uploading Them to HBase

Scenario

This section describes how to use Flume to collect static logs from a local computer (service IP address: 192.168.108.11) and upload them to the **flume_test** table of HBase.

This section applies to MRS 3.*x* or later.

NOTE

By default, the cluster network environment is secure and the SSL authentication is not enabled during the data transmission process. For details about how to use the encryption mode, see **Configuring the Encrypted Transmission**. The configuration can apply to scenarios where only the client or the server is configured, for example, Client/Server:Spooldir Source+File Channel+HBase Sink.

Prerequisites

- The cluster, HBase, and Flume service have been installed.
- The network environment of the cluster is secure.
- An HBase table has been created by running the **create 'flume_test', 'cf'** command.
- The system administrator has understood service requirements and prepared HBase administrator **flume_hbase**.

Procedure

Step 1 On FusionInsight Manager, choose **System > User** and choose **More > Download Authentication Credential** to download the Kerberos certificate file of user **flume_hbase** and save it to the local host.

Step 2 Configure the client parameters of the Flume role.

1. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      Use SpoolDir Source, File Channel, and Avro Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-143** based on the actual environment.

      NOTE

      - If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

   d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-143 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
spoolDir	Specifies the directory where the file to be collected resides. This parameter cannot be left blank. The directory needs to exist and have the write, read, and execute permissions on the flume running user.	/srv/BigData/hadoop/data1/zb
trackerDir	Specifies the path for storing the metadata of files collected by Flume.	/srv/BigData/hadoop/data1/tracker
batchSize	Specifies the number of events that Flume sends in a batch (number of data pieces). A larger value indicates higher performance and lower timeliness.	61200

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint

Parameter	Description	Example Value
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hostname	Specifies the name or IP address of the host whose data is to be sent. This parameter cannot be left blank. Name or IP address must be configured to be the name or IP address that the Avro source associated with it.	192.168.108.11
port	Specifies the port that sends the data. This parameter cannot be left blank. It must be consistent with the port that is monitored by the connected Avro Source.	21154

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

2. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 3 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **server**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.
      For example, use Avro Source, File Channel, and HBase Sink.
   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-144** based on the actual environment.

NOTE

- If the server parameters of the Flume role have been configured, you can choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to non-encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
- A unique checkpoint directory needs to be configured for each File Channel.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-144 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
bind	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as the IP address that the server configuration file will upload.	192.168.108.11
port	Specifies the ID of the port that the Avro Source monitors. This parameter cannot be left blank. It must be configured as an unused port.	21154
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	false

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
table	Specifies the HBase table name. This parameter cannot be left blank.	flume_test
columnFamily	Specifies the HBase column family name. This parameter cannot be left blank.	cf
batchSize	Specifies the maximum number of events written to HBase by Flume in a batch.	61200

Parameter	Description	Example Value
kerberosPrincipal	Specifies the Kerberos authentication user, which is mandatory in security versions. This configuration is required only in security clusters.	flume_hbase
kerberosKeytab	Specifies the file path for Kerberos authentication, which is mandatory in security versions. This configuration is required only in security clusters.	/opt/test/conf/user.keytab **NOTE** Obtain the **user.keytab** file from the Kerberos certificate file of the user **flume_hbase**. In addition, ensure that the user who installs and runs the Flume client has the read and write permissions on the **user.keytab** file.

2. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume**. On the displayed page, click the **Flume** role on the **Instance** tab page.
3. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
4. Click **Save**, and then click **OK**.
5. Click **Finish**.

Step 4 Verify log transmission.

1. Go to the directory where the HBase client is installed.

   cd /*Client installation directory***/ HBase/hbase**

   kinit flume_hbase (Enter the password.)
2. Run the **hbase shell** command to access the HBase client.
3. Run the **scan 'flume_test'** statement. Logs are written in the HBase column family by line.

```
hbase(main):001:0> scan 'flume_test'
ROW                                   COLUMN
+CELL

2017-09-18 16:05:36,394 INFO  [hconnection-0x415a3f6a-shared--pool2-t1] ipc.AbstractRpcClient:
RPC Server Kerberos principal name for service=ClientService is hbase/hadoop.<system domain
name>@<system domain name>
```

```
 default4021ff4a-9339-4151-a4d0-00f20807e76d                column=cf:pCol,
timestamp=1505721909388, value=Welcome to
flume
 incRow                                                     column=cf:iCol, timestamp=1505721909461, value=
\x00\x00\x00\x00\x00\x00\x00\x01
2 row(s) in 0.3660 seconds
```

----End

12.7.11 Encrypted Transmission

12.7.11.1 Configuring the Encrypted Transmission

Scenario

This section describes how to configure the server and client parameters of the Flume service (including the Flume and MonitorServer roles) after the cluster is installed to ensure proper running of the service.

This section applies to MRS 3.*x* or later.

Prerequisites

The cluster and Flume service have been installed.

Procedure

Step 1 Generate the certificate trust lists of the server and client of the Flume role respectively.

1. Remotely log in to the node using ECM where the Flume server is to be installed as user **omm**. Go to the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin** directory.

   cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin

   NOTE

   The version 8.1.0.1 is used as an example. Replace it with the actual version number.

2. Run the following command to generate and export the server and client certificates of the Flume role:

   sh geneJKS.sh -f *xxx* **-g** *xxx*

   The generated certificate is saved in the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf** path .

   - **flume_sChat.jks** is the certificate library of the Flume role server. **flume_sChat.crt** is the exported file of the **flume_sChat.jks** certificate. **-f** indicates the password of the certificate and certificate library.
   - **flume_cChat.jks** is the certificate library of the Flume role client. **flume_cChat.crt** is the exported file of the **flume_cChat.jks** certificate. **-g** indicates the password of the certificate and certificate library.

- **flume_sChatt.jks** and **flume_cChatt.jks** are the SSL certificate trust lists of the Flume server and client, respectively.

NOTE

All user-defined passwords involved in this section must meet the following requirements:

- The password must contain at least four types of uppercase letters, lowercase letters, digits, and special characters.
- The password must contain 8 to 64 characters.
- It is recommended that the user-defined passwords be changed periodically (for example, every three months), and certificates and trust lists be generated again to ensure security.

Step 2 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Remotely log in to any node where the Flume role is located as user **omm** using ECM. Run the following command to go to the ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin directory:

   cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin

2. Run the following command to generate and obtain Flume server keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. It is the password of the **flume_sChat.jks** certificate library.

   ./genPwFile.sh

   cat password.property

3. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.

   a. Log in to FusionInsight Manager. Choose **Services** > **Flume** > **Configuration Tool**.

   b. Set **Agent Name** to **server**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      For example, use Avro Source, File Channel, and HDFS Sink.

   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-145** based on the actual environment.

      NOTE

      - If the server parameters of the Flume role have been configured, you can choose **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Services** > **Flume** > **Import** to change the relevant configuration items of encrypted transmission after the file is imported.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-145 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	true
keystore	Indicates the server certificate.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_sChat.jks
keystore-password	Specifies the password of the key library, which is the password required to obtain the keystore information. Enter the value of password obtained in **Step 2.2**.	-
truststore	Indicates the SSL certificate trust list of the server.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_sChatt.jks
truststore-password	Specifies the trust list password, which is the password required to obtain the truststore information. Enter the value of password obtained in **Step 2.2**.	-

4. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume**. On the displayed page, click the **Flume** role under **Role**.
5. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

NOTE

- An independent server configuration file can be uploaded to each Flume instance.
- This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.

6. Click **Save**, and then click **OK**. Click **Finish**.

Step 3 Set the client parameters of the Flume role.

1. Run the following commands to copy the generated client certificate (**flume_cChat.jks**) and client trust list (**flume_cChatt.jks**) to the client directory, for example, **/opt/flume-client/fusionInsight-flume-1.9.0/conf/**. (The Flume client must have been installed.) **10.196.26.1** is the service plane IP address of the node where the client resides.

   scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_cChat.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

   scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_cChatt.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

   NOTE

   When copying the client certificate, you need to enter the password of user **user** of the host (for example, **10.196.26.1**) where the client resides.

2. Log in to the node where the Flume client is decompressed as user **user**. Run the following command to go to the client directory **opt/flume-client/fusionInsight-flume-1.9.0/bin**.

   cd opt/flume-client/fusionInsight-flume-1.9.0/bin

3. Run the following command to generate and obtain Flume client keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. The password is the same as the password of the certificate whose alias is *flumechatclient* and the password of the *flume_cChat.jks* certificate library.

   ./genPwFile.sh

   cat password.property

   NOTE

   If the following error message is displayed, run the export **JAVA_HOME=*JDK path*** command.

   ```
   JAVA_HOME is null in current user,please install the JDK and set the JAVA_HOME
   ```

4. Run the **echo $SCC_PROFILE_DIR** command to check whether the **SCC_PROFILE_DIR** environment variable is empty.
   - If yes, run the **source .sccfile** command.
   - If no, go to **Step 3.5**.

5. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.

   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.

b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

   For example, use SpoolDir Source, File Channel, and Avro Sink.

c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-146** based on the actual environment.

   NOTE

   - If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to encrypted transmission.
   - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
   - A unique checkpoint directory needs to be configured for each File Channel.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-146 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
ssl	Indicates whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	true
keystore	Specified the client certificate.	/opt/flume-client/fusionInsight-flume-1.9.0/conf/flume_cChat.jks

Parameter	Description	Example Value
keystore-password	Specifies the password of the key library, which is the password required to obtain the keystore information. Enter the value of password obtained in **Step 3.3**.	-
truststore	Indicates the SSL certificate trust list of the client.	/opt/flume-client/fusionInsight-flume-1.9.0/conf/flume_cChatt.jks
truststore-password	Specifies the trust list password, which is the password required to obtain the truststore information. Enter the value of password obtained in **Step 3.3**.	-

6. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 4 Generate the certificate and trust list of the server and client of the MonitorServer role respectively.

1. Log in to the host using ECM with the MonitorServer role assigned as user **omm**.

   Go to the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin** directory.

   cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin

2. Run the following command to generate and export the server and client certificates of the MonitorServer role:

   sh geneJKS.sh -m *xxx* **-n** ***xxx***

   The generated certificate is saved in the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf** path. Where:

   - **ms_sChat.jks** is the certificate library of the MonitorServer role server. **ms_sChat.crt** is the exported file of the **ms_sChat.jks** certificate. **-m** indicates the password of the certificate and certificate library.
   - **ms_cChat.jks** is the certificate library of the MonitorServer role client. **ms_cChat.crt** is the exported file of the **ms_cChat.jks** certificate. **-n** indicates the password of the certificate and certificate library.

- **ms_sChatt.jks** and **ms_cChatt.jks** are the SSL certificate trust lists of the MonitorServer server and client, respectively.

Step 5 Set the server parameters of the MonitorServer role.

1. Run the following command to generate and obtain MonitorServer server keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. The password is the same as the password of the certificate whose alias is *mschatserver* and the password of the *ms_sChat.jks* certificate library.

   ./genPwFile.sh

   cat password.property

2. Run the following command to open the ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/service/application.properties file: Modify related parameters based on the description in **Table 12-147**, save the modification, and exit.

   vi ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1**/install/FusionInsight-Flume-1.9.0/flume/conf/service/application.properties**

   Table 12-147 Parameters to be modified of the MonitorServer role server

Parameter	Description	Example Value
ssl_need_kspasswd_decrypt_key	Specifies whether to enable the user-defined key encryption and decryption function. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_server_enable	Indicates whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_server_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_sChat.jks

Parameter	Description	Example Value
ssl_server_trust_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_sChatt.jks
ssl_server_key_store_password	Indicates the client certificate password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the certificate). Enter the value of password obtained in **Step 5.1**.	-
ssl_server_trust_key_store_password	Specifies the trustkeystore password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the trust list). Enter the value of password obtained in **Step 5.1**.	-
ssl_need_client_auth	Indicates whether to enable the client authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true

3. Restart the MonitorServer instance. Choose **Services** > **Flume** > **Instance** > **MonitorServer**, select the MonitorServer instance, and choose **More** > **Restart Instance**. Enter the administrator password and click **OK**. After the restart is complete, click **Finish**.

Step 6 Set the client parameters of the MonitorServer role.

1. Run the following commands to copy the generated client certificate (**ms_cChat.jks**) and client trust list (**ms_cChatt.jks**) to the **/opt/flume-client/fusionInsight-flume-1.9.0/conf/** client directory. **10.196.26.1** is the service plane IP address of the node where the client resides.

   scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChat.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChatt.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

2. Log in to the node where the Flume client is located as **user**. Run the following command to go to the client directory **/opt/flume-client/fusionInsight-flume-1.9.0/bin**.

   cd /opt/flume-client/fusionInsight-flume-1.9.0/bin

3. Run the following command to generate and obtain MonitorServer client keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. The password is the same as the password of the certificate whose alias is *mschatclient* and the password of the *ms_cChat.jks* certificate library.

   ./genPwFile.sh

   cat password.property

4. Run the following command to open the **/opt/flume-client/fusionInsight-flume-1.9.0/conf/service/application.properties** file. (**/opt/flume-client/fusionInsight-flume-1.9.0** is the directory where the client software is installed.) Modify related parameters based on the description in **Table 12-148**, save the modification, and exit.

   vi /opt/flume-client/fusionInsight-flume-1.9.0/flume/conf/service/application.properties

Table 12-148 Parameters to be modified of the MonitorServer role client

Parameter	Description	Example Value
ssl_need_kspaswd_decrypt_key	Indicates whether to enable the user-defined key encryption and decryption function. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_client_enable	Indicates whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true

Parameter	Description	Example Value
ssl_client_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChat.jks
ssl_client_trust_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChatt.jks
ssl_client_key_store_password	Specifies the keystore password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the certificate). Enter the value of **password** obtained in **Step 6.3**.	-
ssl_client_trust_key_store_password	Specifies the trustkeystore password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the trust list). Enter the value of **password** obtained in **Step 6.3**.	-
ssl_need_client_auth	Indicates whether to enable the client authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true

----End

12.7.11.2 Typical Scenario: Collecting Local Static Logs and Uploading Them to HDFS

Scenario

This section describes how to use Flume to collect static logs from a local PC (service IP address: **192.168.108.11**) and save them to the **/flume/test** directory on HDFS.

This section applies to MRS 3.*x* or later.

Prerequisites

- The cluster, HDFS and Flume services, and Flume client have been installed.
- User **flume_hdfs** has been created, and the HDFS directory and data used for log verification have been authorized to the user.

Procedure

Step 1 Generate the certificate trust lists of the server and client of the Flume role respectively.

1. Log in to the node where the Flume server is located as user **omm**. Go to the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin** directory.

   cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin

2. Run the following command to generate and export the server and client certificates of the Flume role:

   sh geneJKS.sh -f *Password* **-g** *Password*

   The generated certificate is saved in the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf** path .

   - **flume_sChat.jks** is the certificate library of the Flume role server. **flume_sChat.crt** is the exported file of the **flume_sChat.jks** certificate. **-f** indicates the password of the certificate and certificate library.
   - **flume_cChat.jks** is the certificate library of the Flume role client. **flume_cChat.crt** is the exported file of the **flume_cChat.jks** certificate. **-g** indicates the password of the certificate and certificate library.
   - **flume_sChatt.jks** and **flume_cChatt.jks** are the SSL certificate trust lists of the Flume server and client, respectively.

   NOTE

   All user-defined passwords involved in this section must meet the following requirements:

   - Contain at least four types of the following: uppercase letters, lowercase letters, digits, and special characters.
   - Contain at least eight characters and a maximum of 64 characters.
   - It is recommended that the user-defined passwords be changed periodically (for example, every three months), and certificates and trust lists be generated again to ensure security.

Step 2 On FusionInsight Manager, choose **System > User** and choose **More > Download Authentication Credential** to download the Kerberos certificate file of user **flume_hdfs** and save it to the local host.

Step 3 Configure the server parameters of the Flume role and upload the configuration file to the cluster.

1. Log in to any node where the Flume role is located as user **omm**. Run the following command to go to the ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin directory:

   cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin

2. Run the following command to generate and obtain Flume server keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. It is the password of the **flume_sChat.jks** certificate library.

   ./genPwFile.sh

   cat password.property

3. Use the Flume configuration tool on the FusionInsight Manager portal to configure the server parameters and generate the configuration file.

   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.

   b. Set **Agent Name** to **server**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      For example, use Avro Source, File Channel, and HDFS Sink.

   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-149** based on the actual environment.

      NOTE

      - If the server parameters of the Flume role have been configured, you can choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance** on FusionInsight Manager. Then select the corresponding Flume role instance and click the **Download** button behind the **flume.config.file** parameter on the **Instance Configurations** page to obtain the existing server parameter configuration file. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to encrypted transmission.
      - It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.
      - A unique checkpoint directory needs to be configured for each File Channel.

   d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-149 Parameters to be modified of the Flume role server

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
bind	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as the IP address that the server configuration file will upload.	192.168.108.11
port	Specifies the IP address to which Avro Source is bound. This parameter cannot be left blank. It must be configured as an unused port.	21154
ssl	Indicates whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	true
keystore	Indicates the server certificate.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_sChat.jks
keystore-password	Specifies the password of the key library, which is the password required to obtain the keystore information. Enter the value of **password** obtained in **Step 3.2**.	-
truststore	Indicates the SSL certificate trust list of the server.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_sChatt.jks

Parameter	Description	Example Value
truststore-password	Specifies the trust list password, which is the password required to obtain the truststore information. Enter the value of **password** obtained in **Step 3.2**.	-
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flumeserver/checkpoint
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hdfs.path	Specifies the HDFS data write directory. This parameter cannot be left blank.	hdfs://hacluster/flume/test
hdfs.inUsePrefix	Specifies the prefix of the file that is being written to HDFS.	TMP_

Parameter	Description	Example Value
hdfs.batchSize	Specifies the maximum number of events that can be written to HDFS once.	61200
hdfs.kerberosPrincipal	Specifies the Kerberos authentication user, which is mandatory in security versions. This configuration is required only in security clusters.	flume_hdfs
hdfs.kerberosKeytab	Specifies the keytab file path for Kerberos authentication, which is mandatory in security versions. This configuration is required only in security clusters.	/opt/test/conf/user.keytab **NOTE** Obtain the **user.keytab** file from the Kerberos certificate file of the user **flume_hdfs**. In addition, ensure that the user who installs and runs the Flume client has the read and write permissions on the **user.keytab** file.
hdfs.useLocalTimeStamp	Specifies whether to use the local time. Possible values are **true** and **false**.	true

4. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume**. On the displayed page, click the **Flume** role under **Role**.
5. Select the Flume role of the node where the configuration file is to be uploaded, choose **Instance Configurations** > **Import** beside the **flume.config.file**, and select the **properties.properties** file.

   NOTE

   - An independent server configuration file can be uploaded to each Flume instance.
   - This step is required for updating the configuration file. Modifying the configuration file on the background is an improper operation because the modification will be overwritten after configuration synchronization.
6. Click **Save**, and then click **OK**.
7. Click **Finish**.

Step 4 Configure the client parameters of the Flume role.

1. Run the following commands to copy the generated client certificate (**flume_cChat.jks**) and client trust list (**flume_cChatt.jks**) to the client directory, for example, **/opt/flume-client/fusionInsight-flume-1.9.0/conf/**. (The Flume client must have been installed.) **10.196.26.1** is the service plane IP address of the node where the client resides.

scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_cChat.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/flume_cChatt.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

NOTE

When copying the client certificate, you need to enter the password of user **user** of the host (for example, **10.196.26.1**) where the client resides.

2. Log in to the node where the Flume client is decompressed as user **user**. Run the following command to go to the client directory **/opt/flume-client/fusionInsight-flume-1.9.0/bin**.

   cd opt/flume-client/fusionInsight-flume-1.9.0/bin

3. Run the following command to generate and obtain Flume client keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. The password is the same as the password of the certificate whose alias is *flumechatclient* and the password of the *flume_cChat.jks* certificate library.

   ./genPwFile.sh

   cat password.property

   NOTE

   If the following error message is displayed, run the export **JAVA_HOME=*JDKpath*** command.

   ```
   JAVA_HOME is null in current user,please install the JDK and set the JAVA_HOME
   ```

4. Run the **echo $SCC_PROFILE_DIR** command to check whether the **SCC_PROFILE_DIR** environment variable is empty.
   - If yes, run the **source .sccfile** command.
   - If no, go to **Step 4.5**.

5. Use the Flume configuration tool on FusionInsight Manager to configure the Flume role client parameters and generate a configuration file.
   a. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool**.
   b. Set **Agent Name** to **client**. Select the source, channel, and sink to be used, drag them to the GUI on the right, and connect them.

      Use SpoolDir Source, File Channel, and Avro Sink.

   c. Double-click the source, channel, and sink. Set corresponding configuration parameters by seeing **Table 12-150** based on the actual environment.

NOTE

- If the client parameters of the Flume role have been configured, you can obtain the existing client parameter configuration file from *client installation directory*/**fusioninsight-flume-1.9.0/conf/properties.properties** to ensure that the configuration is in concordance with the previous. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Configuration Tool** > **Import**, import the file, and modify the configuration items related to encrypted transmission.
- It is recommended that the numbers of Sources, Channels, and Sinks do not exceed 40 during configuration file import. Otherwise, the response time may be very long.

d. Click **Export** to save the **properties.properties** configuration file to the local.

Table 12-150 Parameters to be modified of the Flume role client

Parameter	Description	Example Value
Name	The value must be unique and cannot be left blank.	test
spoolDir	Specifies the directory where the file to be collected resides. This parameter cannot be left blank. The directory needs to exist and have the write, read, and execute permissions on the flume running user.	/srv/BigData/hadoop/data1/zb
trackerDir	Specifies the path for storing the metadata of files collected by Flume.	/srv/BigData/hadoop/data1/tracker
batch-size	Specifies the number of events that Flume sends in a batch.	61200

Parameter	Description	Example Value
dataDirs	Specifies the directory for storing buffer data. The run directory is used by default. Configuring multiple directories on disks can improve transmission efficiency. Use commas (,) to separate multiple directories. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/data** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/data
checkpointDir	Specifies the directory for storing the checkpoint information, which is under the run directory by default. If the directory is inside the cluster, the **/srv/BigData/hadoop/dataX/flume/checkpoint** directory can be used. **dataX** ranges from data1 to dataN. If the directory is outside the cluster, it needs to be independently planned.	/srv/BigData/hadoop/data1/flume/checkpoint

Parameter	Description	Example Value
transactionCapacity	Specifies the transaction size, that is, the number of events in a transaction that can be processed by the current Channel. The size cannot be smaller than the batchSize of Source. Setting the same size as batchSize is recommended.	61200
hostname	Specifies the name or IP address of the host whose data is to be sent. This parameter cannot be left blank. Name or IP address must be configured to be the name or IP address that the Avro source associated with it.	192.168.108.11
port	Specifies the IP address to which Avro Sink is bound. This parameter cannot be left blank. It must be consistent with the port that is monitored by the connected Avro Source.	21154

Parameter	Description	Example Value
ssl	Specifies whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) Only Sources of the Avro type have this configuration item. ■ **true** indicates that the function is enabled. ■ **false** indicates that the client authentication function is not enabled.	true
keystore	Specifies the **flume_cChat.jks** certificate generated on the server.	/opt/flume-client/fusionInsight-flume-1.9.0/conf/flume_cChat.jks
keystore-password	Specifies the password of the key library, which is the password required to obtain the keystore information. Enter the value of **password** obtained in **Step 4.3**.	-
truststore	Indicates the SSL certificate trust list of the server.	/opt/flume-client/fusionInsight-flume-1.9.0/conf/flume_cChatt.jks
truststore-password	Specifies the trust list password, which is the password required to obtain the truststore information. Enter the value of **password** obtained in **Step 4.3**.	-

6. Upload the **properties.properties** file to **flume/conf/** under the installation directory of the Flume client.

Step 5 Generate the certificate and trust list of the server and client of the MonitorServer role respectively.

1. Log in to the host with the MonitorServer role assigned as user **omm**.

   Go to the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin** directory.

   cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/bin

2. Run the following command to generate and export the server and client certificates of the MonitorServer role:

   sh geneJKS.sh -m *Password* **-n** *Password*

   The generated certificate is saved in the **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf** path. Where:

   - **ms_sChat.jks** is the certificate library of the MonitorServer role server. **ms_sChat.crt** is the exported file of the **ms_sChat.jks** certificate. **-m** indicates the password of the certificate and certificate library.
   - **ms_cChat.jks** is the certificate library of the MonitorServer role client. **ms_cChat.crt** is the exported file of the **ms_cChat.jks** certificate. **-n** indicates the password of the certificate and certificate library.
   - **ms_sChatt.jks** and **ms_cChatt.jks** are the SSL certificate trust lists of the MonitorServer server and client, respectively.

Step 6 Set the server parameters of the MonitorServer role.

1. Run the following command to generate and obtain MonitorServer server keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. The password is the same as the password of the certificate whose alias is *mschatserver* and the password of the *ms_sChat.jks* certificate library.

   ./genPwFile.sh

   cat password.property

2. Run the following command to open the ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/service/application.properties file: Modify related parameters based on the description in **Table 12-151**, save the modification, and exit.

   vi ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1**/install/FusionInsight-Flume-1.9.0/flume/conf/service/application.properties**

Table 12-151 Parameters to be modified of the MonitorServer role server

Parameter	Description	Example Value
ssl_need_kspasswd_decrypt_key	Indicates whether to enable the user-defined key encryption and decryption function. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_server_enable	Indicates whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_server_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_sChat.jks
ssl_server_trust_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_sChatt.jks
ssl_server_key_store_password	Indicates the client certificate password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the certificate). Enter the value of **password** obtained in **Step 6.1**.	-

Parameter	Description	Example Value
ssl_server_trust_key_store_password	Indicates the client trust list password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the trust list). Enter the value of **password** obtained in **Step 6.1**.	-
ssl_need_client_auth	Indicates whether to enable the client authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true

3. Restart the MonitorServer instance. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Instance** > **MonitorServer**, select the configured MonitorServer instance, and choose **More** > **Restart Instance**. Enter the administrator password and click **OK**. After the restart is complete, click **Finish**.

Step 7 Set the client parameters of the MonitorServer role.

1. Run the following commands to copy the generated client certificate (**ms_cChat.jks**) and client trust list (**ms_cChatt.jks**) to the **/opt/flume-client/fusionInsight-flume-1.9.0/conf/** client directory. **10.196.26.1** is the service plane IP address of the node where the client resides.

   scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChat.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

   scp ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChatt.jks user@10.196.26.1:/opt/flume-client/fusionInsight-flume-1.9.0/conf/

2. Log in to the node where the Flume client is located as user **user**. Run the following command to go to the client directory **/opt/flume-client/fusionInsight-flume-1.9.0/bin**.

   cd /opt/flume-client/fusionInsight-flume-1.9.0/bin

3. Run the following command to generate and obtain MonitorServer client keystore password, trust list password, and keystore-password encrypted private key information. Enter the password twice and confirm the password. The password is the same as the password of the certificate whose alias is *mschatclient* and the password of the *ms_cChat.jks* certificate library.

./genPwFile.sh

cat password.property

4. Run the following command to open the **/opt/flume-client/fusionInsight-flume-1.9.0/conf/service/application.properties** file. (**/opt/flume-client/fusionInsight-flume-1.9.0** is the directory where the client is installed.) Modify related parameters based on the description in **Table 12-152**, save the modification, and exit.

   vi /opt/flume-client/fusionInsight-flume-1.9.0/conf/service/application.properties

   Table 12-152 Parameters to be modified of the MonitorServer role client

Parameter	Description	Example Value
ssl_need_kspasswd_decrypt_key	Indicates whether to enable the user-defined key encryption and decryption function. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_client_enable	Indicates whether to enable the SSL authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true
ssl_client_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChat.jks
ssl_client_trust_key_store	Set this parameter based on the specific storage location.	${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Flume-1.9.0/flume/conf/ms_cChatt.jks

Parameter	Description	Example Value
ssl_client_key_store_password	Specifies the keystore password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the certificate). Enter the value of **password** obtained in **Step 7.3**.	-
ssl_client_trust_key_store_password	Specifies the trustkeystore password. Set this parameter based on the actual situation of certificate creation (the plaintext key used to generate the trust list). Enter the value of **password** obtained in **Step 7.3**.	-
ssl_need_client_auth	Indicates whether to enable the client authentication. (You are advised to enable this function to ensure security.) – **true** indicates that the function is enabled. – **false** indicates that the client authentication function is not enabled.	true

Step 8 Verify log transmission.

1. Log in to FusionInsight Manager as a user who has the management permission on HDFS. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS**, click the HDFS WebUI link to go to the HDFS WebUI, and choose **Utilities** > **Browse the file system**.
2. Check whether the data is generated in the **/flume/test** directory on the HDFS.

----End

12.7.12 Viewing Flume Client Monitoring Information

Scenario

The Flume client outside the FusionInsight cluster is a part of the end-to-end data collection. Both the Flume client outside the cluster and the Flume server in the cluster need to be monitored. Users can use FusionInsight Manager to monitor the Flume client and view the monitoring indicators of the Source, Sink, and Channel of the client as well as the client process status.

This section applies to MRS 3.*x* or later.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Flume** > **Flume Management** to view the current Flume client list and process status.

Step 3 Click the **Instance ID**, and view client monitoring metrics in the **Current** area.

Step 4 Click **History**. The page for querying historical monitoring data is displayed. Select a time range and click **View** to view the monitoring data within the time range.

----End

12.7.13 Connecting Flume to Kafka in Security Mode

Scenario

This section describes how to connect to Kafka using the Flume client in security mode.

This section applies to MRS 3.*x* or later.

Procedure

Step 1 Create a **jaas.conf** file and save it to **${***Flume client installation directory***} /conf**. The content of the **jaas.conf** file is as follows:

```
KafkaClient {
com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
keyTab="/opt/test/conf/user.keytab"
principal="flume_hdfs@<System domain name>"
useTicketCache=false
storeKey=true
debug=true;
 };
```

Set **keyTab** and **principal** based on site requirements. The configured **principal** must have certain kafka permissions.

Step 2 Configure services. Set the port number of **kafka.bootstrap.servers** to **21007**, and set **kafka.security.protocol** to **SASL_PLAINTEXT**.

Step 3 If the domain name of the cluster where Kafka is located is changed, change the value of *-Dkerberos.domain.name* in the **flume-env.sh** file in **$**{*Flume client installation directory*} **/conf/** based on the site requirements.

Step 4 Upload the configured **properties.properties** file to **${***Flume client installation directory***} /conf**.

----End

12.7.14 Connecting Flume with Hive in Security Mode

Scenario

This section describes how to use Flume to connect to Hive (version 3.1.0) in the cluster.

This section applies to MRS 3.*x* or later.

Prerequisites

Flume and Hive have been correctly installed in the cluster. The services are running properly, and no alarm is reported.

Procedure

Step 1 Import the following JAR packages to the lib directory (client/server) of the Flume instance to be tested as user **omm**:

- antlr-2.7.7.jar
- antlr-runtime-3.4.jar
- calcite-core-1.16.0.jar
- hadoop-mapreduce-client-core-3.1.1.jar
- hive-beeline-3.1.0.jar
- hive-cli-3.1.0.jar
- hive-common-3.1.0.jar
- hive-exec-3.1.0.jar
- hive-hcatalog-core-3.1.0.jar
- hive-hcatalog-***-adapter-3.1.0.jar
- hive-hcatalog-server-extensions-3.1.0.jar
- hive-hcatalog-streaming-3.1.0.jar
- hive-metastore-3.1.0.jar
- hive-service-3.1.0.jar
- libfb303-0.9.3.jar
- hadoop-plugins-1.0.jar

You can obtain the JAR package from the Hive installation directory and restart the Flume process to ensure that the JAR package is loaded to the running environment.

Step 2 Set Hive configuration items.

On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations** > **HiveServer** > **Customization** > **hive.server.customized.configs**.

Example configurations:

Name	Value
hive.support.concurrency	true
hive.exec.dynamic.partition.mode	nonstrict
hive.txn.manager	org.apache.hadoop.hive.ql.lockmgr.DbTxnManager
hive.compactor.initiator.on	true
hive.compactor.worker.threads	1

Step 3 Prepare the system user **flume_hive** who has the supergroup and Hive permissions, install the client, and create the required Hive table.

Example:

1. The cluster client has been correctly installed. For example, the installation directory is **/opt/client**.
2. Run the following command to authenticate the user:

   cd /opt/client

   source bigdata_env

   kinit flume_hive
3. Run the **beeline** command and run the following table creation statement:

   ```
   create table flume_multi_type_part(id string, msg string)
   partitioned by (country string, year_month string, day string)
   clustered by (id) into 5 buckets
   stored as orc TBLPROPERTIES('transactional'='true');
   ```
4. Run the **select * from** *Table name***;** command to query data in the table.

   In this case, the number of data records in the table is **0**.

Step 4 Prepare related configuration files. Assume that the client installation package is stored in **/opt/FusionInsight_Cluster_1_Services_ClientConfig**.

1. Obtain the following files from the $*Client decompression directory*/**Hive/config** directory:
   - hivemetastore-site.xml
   - hive-site.xml
2. Obtain the following files from the **$***Client decompression directory*/**HDFS/config** directory:

   core-site.xml
3. Create a directory on the host where the Flume instance is started and save the prepared files to the created directory.

   Example: **/opt/hivesink-conf/hive-site.xml**.
4. Copy all property configurations in the **hivemetastore-site.xml** file to the **hive-site.xml** file and ensure that the configurations are placed before the original configurations.

   Data is loaded in sequence in Hive.

NOTE

Ensure that the Flume running user **omm** has the read and write permissions on the directory where the configuration file is stored.

Step 5 Observe the result.

On the Hive client, run the **select * from** *Table name*; command. Check whether the corresponding data has been written to the Hive table.

----End

Examples

Flume configuration example (SpoolDir--Mem--Hive):

```
server.sources = spool_source
server.channels = mem_channel
server.sinks = Hive_Sink

#config the source
server.sources.spool_source.type = spooldir
server.sources.spool_source.spoolDir = /tmp/testflume
server.sources.spool_source.montime =
server.sources.spool_source.fileSuffix =.COMPLETED
server.sources.spool_source.deletePolicy = never
server.sources.spool_source.trackerDir =.flumespool
server.sources.spool_source.ignorePattern = ^$
server.sources.spool_source.batchSize = 20
server.sources.spool_source.inputCharset =UTF-8
server.sources.spool_source.selector.type = replicating
server.sources.spool_source.fileHeader = false
server.sources.spool_source.fileHeaderKey = file
server.sources.spool_source.basenameHeaderKey= basename
server.sources.spool_source.deserializer = LINE
server.sources.spool_source.deserializer.maxBatchLine= 1
server.sources.spool_source.deserializer.maxLineLength= 2048
server.sources.spool_source.channels = mem_channel

#config the channel
server.channels.mem_channel.type = memory
server.channels.mem_channel.capacity =10000
server.channels.mem_channel.transactionCapacity= 2000
server.channels.mem_channel.channelfullcount= 10
server.channels.mem_channel.keep-alive = 3
server.channels.mem_channel.byteCapacity =
server.channels.mem_channel.byteCapacityBufferPercentage= 20

#config the sink
server.sinks.Hive_Sink.type = hive
server.sinks.Hive_Sink.channel = mem_channel
server.sinks.Hive_Sink.hive.metastore = thrift://${any MetaStore service IP address}:21088
server.sinks.Hive_Sink.hive.hiveSite = /opt/hivesink-conf/hive-site.xml
server.sinks.Hive_Sink.hive.coreSite = /opt/hivesink-conf/core-site.xml
server.sinks.Hive_Sink.hive.metastoreSite = /opt/hivesink-conf/hivemeatastore-site.xml
server.sinks.Hive_Sink.hive.database = default
server.sinks.Hive_Sink.hive.table = flume_multi_type_part
server.sinks.Hive_Sink.hive.partition = Tag,%Y-%m,%d
server.sinks.Hive_Sink.hive.txnsPerBatchAsk= 100
server.sinks.Hive_Sink.hive.autoCreatePartitions= true
server.sinks.Hive_Sink.useLocalTimeStamp = true
server.sinks.Hive_Sink.batchSize = 1000
server.sinks.Hive_Sink.hive.kerberosPrincipal= super1
server.sinks.Hive_Sink.hive.kerberosKeytab= /opt/mykeytab/user.keytab
server.sinks.Hive_Sink.round = true
server.sinks.Hive_Sink.roundValue = 10
server.sinks.Hive_Sink.roundUnit = minute
server.sinks.Hive_Sink.serializer = DELIMITED
```

```
server.sinks.Hive_Sink.serializer.delimiter= ";"
server.sinks.Hive_Sink.serializer.serdeSeparator= ';'
server.sinks.Hive_Sink.serializer.fieldnames= id,msg
```

12.7.15 Configuring the Flume Service Model

12.7.15.1 Overview

This section applies to MRS 3.*x* or later.

Guide a reasonable Flume service configuration by providing performance differences between Flume common modules, to avoid a nonstandard overall service performance caused when a frontend Source and a backend Sink do not match in performance.

Only single channels are compared for description.

12.7.15.2 Service Model Configuration Guide

This section applies to MRS 3.*x* or later.

During Flume service configuration and module selection, the ultimate throughput of a sink must be greater than the maximum throughput of a source. Otherwise, in extreme load scenarios, the write speed of the source to a channel is greater than the read speed of sink from channel. Therefore, the channel is fully occupied due to frequent usage, and the performance is affected.

Avro Source and Avro Sink are usually used in pairs to transfer data between multiple Flume Agents. Therefore, Avro Source and Avro Sink do not become a performance bottleneck in general scenarios.

Inter-Module Performance

Based on comparison between the limit performances of modules, Kafka Sink and HDFS Sink can meet the throughput requirements when the front-end is SpoolDir Source. However, HBase Sink could become performance bottlenecks due to the low write performances thereof. As a result, data is stacked in Channel. If you have to use HBase Sink or other sinks that are prone to become performance bottlenecks, you can use **Channel Selector** or **Sink Group** to meet performance requirements.

Channel Selector

A channel selector allows a source to connect to multiple channels. Data of the source can be distributed or copied by selecting different types of selectors. Currently, a channel selector provided by Flume can be a replicating channel selector or a multiplexing channel selector.

Replicating: indicates that the data of the source is synchronized to all channels.

Multiplexing: indicates that based on the value of a specific field of the header of an event, a channel is selected to send the data. In this way, the data is distributed based on a service type.

- Replicating configuration example:

```
client.sources = kafkasource
client.channels = channel1 channel2
client.sources.kafkasource.type = org.apache.flume.source.kafka.KafkaSource
client.sources.kafkasource.kafka.topics = topic1,topic2
client.sources.kafkasource.kafka.consumer.group.id = flume
client.sources.kafkasource.kafka.bootstrap.servers = 10.69.112.108:21007
client.sources.kafkasource.kafka.security.protocol = SASL_PLAINTEXT
client.sources.kafkasource.batchDurationMillis = 1000
client.sources.kafkasource.batchSize = 800
client.sources.kafkasource.channels = channel1 c el2

client.sources.kafkasource.selector.type = replicating
client.sources.kafkasource.selector.optional = channel2
```

Table 12-153 Parameters in the Replicating configuration example

Parameter	Default Value	Description
Selector.type	replicating	Selector type. Set this parameter to **replicating**.
Selector.optional	-	Optional channel. Configure this parameter as a list.

- Multiplexing configuration example:

```
client.sources = kafkasource
client.channels = channel1 channel2
client.sources.kafkasource.type = org.apache.flume.source.kafka.KafkaSource
client.sources.kafkasource.kafka.topics = topic1,topic2
client.sources.kafkasource.kafka.consumer.group.id = flume
client.sources.kafkasource.kafka.bootstrap.servers = 10.69.112.108:21007
client.sources.kafkasource.kafka.security.protocol = SASL_PLAINTEXT
client.sources.kafkasource.batchDurationMillis = 1000
client.sources.kafkasource.batchSize = 800
client.sources.kafkasource.channels = channel1 channel2

client.sources.kafkasource.selector.type = multiplexing
client.sources.kafkasource.selector.header = myheader
client.sources.kafkasource.selector.mapping.topic1 = channel1
client.sources.kafkasource.selector.mapping.topic2 = channel2
client.sources.kafkasource.selector.default = channel1
```

Table 12-154 Parameters in the Multiplexing configuration example

Parameter	Default Value	Description
Selector.type	replicating	Selector type. Set this parameter to **multiplexing**.
Selector.header	Flume.selector.header	-
Selector.default	-	-
Selector.mapping.*	-	-

In a multiplexing selector example, select a field whose name is topic from the header of the event. When the value of the topic field in the header is

topic1, send the event to a channel 1; or when the value of the topic field in the header is topic2, send the event to a channel 2.

Selectors need to use a specific header of an event in a source to select a channel, and need to select a proper header based on a service scenario to distribute data.

SinkGroup

When the performance of a backend single sink is insufficient, and high reliability or heterogeneous output is required, you can use a sink group to connect a specified channel to multiple sinks, thereby meeting use requirements. Currently, Flume provides two types of sink processors to manage sinks in a sink group. The types are load balancing and failover.

Failover: Indicates that there is only one active sink in the sink group each time, and the other sinks are on standby and inactive. When the active sink becomes faulty, one of the inactive sinks is selected based on priorities to take over services, so as to ensure that data is not lost. This is used in high-reliability scenarios.

Load balancing: Indicates that all sinks in the sink group are active. Each sink obtains data from the channel and processes the data. In addition, during running, loads of all sinks in the sink group are balanced. This is used in performance improvement scenarios.

- Load balancing configuration examples:

```
client.sources = source1
client.sinks = sink1 sink2
client.channels = channel1

client.sinkgroups = g1
client.sinkgroups.g1.sinks = sink1 sink2
client.sinkgroups.g1.processor.type = load_balance
client.sinkgroups.g1.processor.backoff = true
client.sinkgroups.g1.processor.selector = random

client.sinks.sink1.type = logger
client.sinks.sink1.channel = channel1

client.sinks.sink2.type = logger
client.sinks.sink2.channel = channel1
```

Table 12-155 Parameters of Load Balancing configuration examples

Parameter	Default Value	Description
sinks	-	Specifies the sink list of the sink group. Multiple sinks are separated by spaces.
processor.type	default	Specifies the type of a processor. Set this parameter to **load_balance**.
processor.backoff	false	Indicates whether to back off failed sinks exponentially.

Parameter	Default Value	Description
processor.selector	round_robin	Specifies the selection mechanism. It must be round_robin, random, or a customized class that inherits AbstractSinkSelector.
processor.selector.maxTimeOut	30000	Specifies the time for masking a faulty sink. The default value is 30,000 ms.

- Failover configuration examples:

```
client.sources = source1
client.sinks = sink1 sink2
client.channels = channel1

client.sinkgroups = g1
client.sinkgroups.g1.sinks = sink1 sink2
client.sinkgroups.g1.processor.type = failover
client.sinkgroups.g1.processor.priority.sink1 = 10
client.sinkgroups.g1.processor.priority.sink2 = 5
client.sinkgroups.g1.processor.maxpenalty = 10000

client.sinks.sink1.type = logger
client.sinks.sink1.channel = channel1

client.sinks.sink2.type = logger
client.sinks.sink2.channel = channel1
```

Table 12-156 Parameters in the **failover** configuration example

Parameter	Default Value	Description
sinks	-	Specifies the sink list of the sink group. Multiple sinks are separated by spaces.
processor.type	default	Specifies the type of a processor. Set this parameter to **failover**.

Parameter	Default Value	Description
processor.priority.<sinkName>	-	Priority. **<sinkName>** must be defined in description of sinks. A sink having a higher priority is activated earlier. A larger value indicates a higher priority. **Note**: If there are multiple sinks, their priorities must be different. Otherwise, only one of them takes effect.
processor.maxpenalty	30000	Specifies the maximum backoff time of failed sinks (unit: ms).

Interceptors

The Flume interceptor supports modification or discarding of basic unit events during data transmission. You can specify the class name list of built-in interceptors in Flume or develop customized interceptors to modify or discard events. The following table lists the built-in interceptors in Flume. A complex example is used in this section. Other users can configure and use interceptions as required. For details, visit the following website:

http://flume.apache.org/releases/content/1.9.0/FlumeUserGuide.html

NOTE

1. The interceptor is used between the sources and channels of Flume. Most sources provide parameters for configuring interceptors. You can set the parameters as required.
2. Flume allows multiple interceptors to be configured for a source. The interceptor names are separated by spaces.
3. The specified interceptor sequence is the order in which they are called.
4. The contents inserted by the interceptor in the header can be read and used in sink.

Table 12-157 Types of built-in interceptors in Flume

Interceptor Type	Description
Timestamp Interceptor	The interceptor inserts a timestamp into the header of an event.
Host Interceptor	The interceptor inserts the IP address or host name of the node where the agent is located into the Header of an event.

Interceptor Type	Description
Remove Header Interceptor	The interceptor discards the corresponding event based on the strings that matches the regular expression contained in the event header.
UUID Interceptor	The interceptor generates a UUID string for the header of each event.
Search and Replace Interceptor	The interceptor provides a simple string-based search and replacement function based on Java regular expressions. The rule is the same as that of Java Matcher.replaceAll().
Regex Filtering Interceptor	The interceptor uses the body of an event as a text file and matches the configured regular expression to filter events. The provided regular expression can be used to exclude or include events.
Regex Extractor Interceptor	The interceptor extracts content from the original events using a regular expression and adds the content to the header of events.

Regex Filtering Interceptor is used as an example to describe how to use the interceptor. (For other types of interceptions, see the configuration provided on the official website.)

Table 12-158 Parameter configuration for **Regex Filtering Interceptor**

Parameter	Default Value	Description
type	-	Specifies the component type name. The value must be **regex_filter**.
regex	-	Specifies the regular expression used to match events.
excludeEvents	false	By default, the matched events are collected. If this parameter is set to **true**, the matched events are deleted and the unmatched events are retained.

Configuration example (netcat tcp is used as the source, and logger is used as the sink). After configuring the preceding parameters, run the **telnet** *Host name or IP address* **44444** command on the host where the Linux operating system is run, and enter a string that complies with the regular expression and another does not

comply with the regular expression. The log shows that only the matched string is transmitted.

```
#define the source, channel, sink
server.sources = r1

server.channels = c1
server.sinks = k1

#config the source
server.sources.r1.type = netcat
server.sources.r1.bind = ${Host IP address}
server.sources.r1.port = 44444
server.sources.r1.interceptors= i1
server.sources.r1.interceptors.i1.type= regex_filter
server.sources.r1.interceptors.i1.regex= (flume)|(myflume)
server.sources.r1.interceptors.i1.excludeEvents= false
server.sources.r1.channels = c1

#config the channel
server.channels.c1.type = memory
server.channels.c1.capacity = 1000
server.channels.c1.transactionCapacity = 100
#config the sink
server.sinks.k1.type = logger
server.sinks.k1.channel = c1
```

12.7.16 Introduction to Flume Logs

Log Description

Log path: The default path of Flume log files is **/var/log/Bigdata/***Role name*.

- FlumeServer: **/var/log/Bigdata/flume/flume**
- FlumeClient: **/var/log/Bigdata/flume-client-n/flume**
- MonitorServer: **/var/log/Bigdata/flume/monitor**

Log archive rule: The automatic Flume log compression function is enabled. By default, when the size of logs exceeds 50 MB , logs are automatically compressed into a log file named in the following format: *<Original log file name>-<yyyy-mm-dd_hh-mm-ss>.[ID]***.log.zip**. A maximum of 20 latest compressed files are reserved. The number of compressed files can be configured on the Manager portal.

Table 12-159 Flume log list

Type	Name	Description
Run logs	/flume/flumeServer.log	Log file that records FlumeServer running environment information.
	/flume/install.log	FlumeServer installation log file
	/flume/flumeServer-gc.log.*<No.>*	GC log file of the FlumeServer process
	/flume/prestartDvietail.log	Work log file before the FlumeServer startup

Type	Name	Description
	/flume/startDetail.log	Startup log file of the Flume process
	/flume/stopDetail.log	Shutdown log file of the Flume process
	/monitor/monitorServer.log	Log file that records MonitorServer running environment information
	/monitor/startDetail.log	Startup log file of the MonitorServer process
	/monitor/stopDetail.log	Shutdown log file of the MonitorServer process
	function.log	External function invoking log file

Log Level

Table 12-160 describes the log levels supported by Flume.

Levels of run logs are FATAL, ERROR, WARN, INFO, and DEBUG from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-160 Log level

Type	Level	Description
Run log	FATAL	Logs of this level record critical error information about system running.
	ERROR	Logs of this level record error information about system running.
	WARN	Logs of this level record exception information about the current event processing.
	INFO	Logs of this level record normal running status information about the system and events.

Type	Level	Description
	DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of Flume by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

----End

NOTE

The configurations take effect immediately without the need to restart the service.

Log Format

The following table lists the Flume log formats.

Table 12-161 Log format

Type	Format	Example
Run logs	*<yyyy-MM-dd HH:mm:ss,SSS>|<Log level>|<Name of the thread that generates the log>|<Message in the log>|<Location where the log event occurs>*	2014-12-12 11:54:57,316 | INFO | [main] | log4j dynamic load is start. | org.apache.flume.tools.LogDynamicLoad.start(LogDynamicLoad.java:59)
	<yyyy-MM-dd HH:mm:ss,SSS><Username><User IP><Time><Operation><Resource><Result><Detail>	2014-12-12 23:04:16,572 | INFO | [SinkRunner-PollingRunner-DefaultSinkProcessor] | SRCIP=null OPERATION=close

12.7.17 Flume Client Cgroup Usage Guide

Scenario

This section describes how to join and log out of a cgroup, query the cgroup status, and change the cgroup CPU threshold.

This section applies to MRS 3.*x* or later.

Procedure

- **Join Cgroup**

  Assume that the Flume client installation path is **/opt/FlumeClient**, and the cgroup CPU threshold is 50%. Run the following command to join a cgroup:

  cd /opt/FlumeClient/fusioninsight-flume-1.9.0/bin

  ./flume-manage.sh cgroup join 50

  NOTE

  - This command can be used to join a cgroup and change the cgroup CPU threshold.
  - The value of the CPU threshold of a cgroup ranges from 1 to 100 x *N*. *N* indicates the number of CPU cores.

- **Check Cgroup status**

  Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following commands to query the cgroup status:

  cd /opt/FlumeClient/fusioninsight-flume-1.9.0/bin

  ./flume-manage.sh cgroup status

- **Exit Cgroup**

  Assume that the Flume client installation path is **/opt/FlumeClient**. Run the following commands to exit cgroup:

  cd /opt/FlumeClient/fusioninsight-flume-1.9.0/bin

  ./flume-manage.sh cgroup exit

  NOTE

  - After the client is installed, the default cgroup is automatically created. If the **-s** parameter is not configured during client installation, the default value **-1** is used. The default value indicates that the agent process is not restricted by the CPU usage.
  - Joining or exiting a cgroup does not affect the agent process. Even if the agent process is not started, the joining or exiting operation can be performed successfully, and the operation will take effect after the next startup of the agent process.
  - After the client is uninstalled, the cgroups created during the client installation are automatically deleted.

12.7.18 Secondary Development Guide for Flume Third-Party Plug-ins

Scenario

This section describes how to perform secondary development for third-party plug-ins.

This section applies to MRS 3.*x* or later.

Prerequisites

- You have obtained the third-party JAR package.
- You have installed Flume server or client.

Procedure

Step 1 Compress the self-developed code into a JAR package.

Step 2 Create a directory for the plug-in.

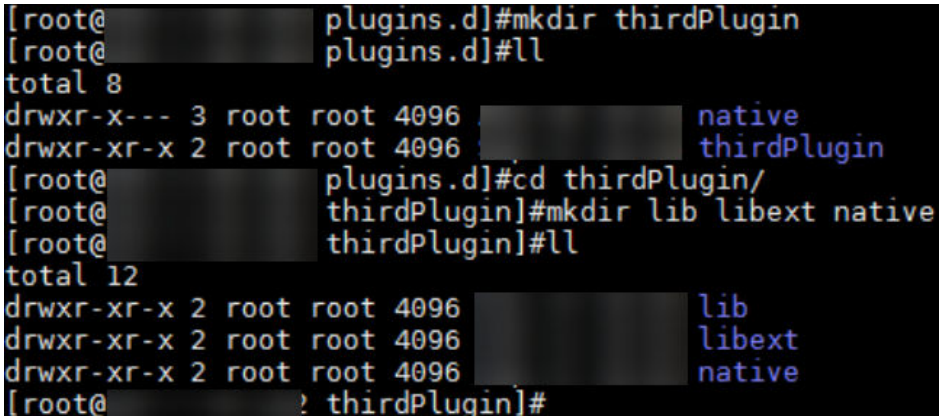
1. Access the **$FLUME_HOME/plugins.d** path and run the following command to create a directory:

   mkdir thirdPlugin

   cd thirdPlugin

   mkdir lib libext native

   The command output is displayed as follows:

2. Place the third-party JAR package in the **$FLUME_HOME/plugins.d/thirdPlugin/lib** directory. If the JAR package depends on other JAR packages, place the depended JAR packages to the **$FLUME_HOME/ plugins.d/thirdPlugin/libext** directory, and place the local library files in **$FLUME_HOME/ plugins.d/ thirdPlugin/native**.

Step 3 Configure the **properties.properties** file in **$FLUME_HOME/conf/**.

For details about how to set parameters in the **properties.properties** file, see the parameter list in the **properties.properties** file in the corresponding typical scenario **Non-Encrypted Transmission** and **Encrypted Transmission**.

NOTE

- **$FLUME_HOME** indicates the Flume installation path. Set this parameter based on the site requirements (server or client) when configuring third-party plug-ins.
- **thirdPlugin** is the name of the third-party plugin.

----End

12.7.19 Common Issues About Flume

Flume logs are stored in **/var/log/Bigdata/flume/flume/flumeServer.log**. Most data transmission exceptions and data transmission failures are recorded in logs. You can run the following command:

tailf /var/log/Bigdata/flume/flume/flumeServer.log

- Problem: After the configuration file is uploaded, an exception occurs. After the configuration file is uploaded again, the scenario requirements are still not met, but no exception is recorded in the log.

  Solution: Restart the Flume process, run the **kill -9** *Process code* to kill the process code, and view the logs.

- Issue: **"java.lang.IllegalArgumentException: Keytab is not a readable file: /opt/test/conf/user.keytab"** is displayed when HDFS is connected.

  Solution: Grant the read and write permissions to the Flume running user.

- Problem: The following error is reported when the Flume client is connected to Kafka:

  ```
  Caused by: java.io.IOException: /opt/FlumeClient/fusioninsight-flume-1.9.0/cof//jaas.conf (No such file or directory)
  ```

  Solution: Add the **jaas.conf** configuration file and save it to the **conf** directory of the Flume client.

  vi jaas.conf

  ```
  KafkaClient {
  com.sun.security.auth.module.Krb5LoginModule required
  useKeyTab=true
  keyTab="/opt/test/conf/user.keytab"
  principal="flume_hdfs@<System domain name>"
  useTicketCache=false
  storeKey=true
  debug=true;
  };
  ```

  Values of **keyTab** and **principal** vary depending on the actual situation.

- Problem: The following error is reported when the Flume client is connected to HBase:

  ```
  Caused by: java.io.IOException: /opt/FlumeClient/fusioninsight-flume-1.9.0/cof//jaas.conf (No such file or directory)
  ```

  Solution: Add the **jaas.conf** configuration file and save it to the **conf** directory of the Flume client.

  vi jaas.conf

  ```
  Client {
  com.sun.security.auth.module.Krb5LoginModule required
  useKeyTab=true
  keyTab="/opt/test/conf/user.keytab"
  principal="flume_hbase@<System domain name>"
  useTicketCache=false
  storeKey=true
  ```

```
debug=true;
};
```

Values of **keyTab** and **principal** vary depending on the actual situation.

- Question: After the configuration file is submitted, the Flume Agent occupies resources. How do I restore the Flume Agent to the state when the configuration file is not uploaded?

  Solution: Submit an empty **properties.properties** file.

12.8 Using HBase

12.8.1 Using HBase from Scratch

HBase is a column-based distributed storage system that features high reliability, performance, and scalability. This section describes how to use HBase from scratch, including how to update the client on the Master node in the cluster, create a table using the client, insert data in the table, modify the table, read data from the table, delete table data, and delete the table.

Background

Suppose a user develops an application to manage users who use service A in an enterprise. The procedure of operating service A on the HBase client is as follows:

- Create the **user_info** table.
- Add users' educational backgrounds and titles to the table.
- Query user names and addresses by user ID.
- Query information by user name.
- Deregister users and delete user data from the user information table.
- Delete the user information table after service A ends.

Table 12-162 User information

ID	Name	Gender	Age	Address
12005000201	A	Male	19	City A
12005000202	B	Female	23	City B
12005000203	C	Male	26	City C
12005000204	D	Male	18	City D
12005000205	E	Female	21	City E
12005000206	F	Male	32	City F
12005000207	G	Female	29	City G
12005000208	H	Female	30	City H
12005000209	I	Male	26	City I

ID	Name	Gender	Age	Address
12005000210	J	Male	25	City J

Prerequisites

The client has been installed. For example, the client is installed in the **/opt/client** directory. The client directory in the following operations is only an example. Change it to the actual installation directory. Before using the client, download and update the client configuration file, and ensure that the active management node of Manager is available.

Procedure

For versions earlier than MRS 3.x, perform the following operations:

Step 1 Download the client configuration file.

1. Log in to MRS Manager. For details, see **Accessing Manager**. Then, choose **Services**.
2. Click **Download Client**.

   Set **Client Type** to **Only configuration files**, **Download To** to **Server**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default. You can customize the file path.

Step 2 Log in to the active management node of MRS Manager.

1. On the **Node** tab page, view the **Name** parameter. The node that contains **master1** in its name is the Master1 node. The node that contains **master2** in its name is the Master2 node.

   The active and standby management nodes of MRS Manager are installed on Master nodes by default. Because Master1 and Master2 are switched over in active and standby mode, Master1 is not always the active management node of MRS Manager. Run a command in Master1 to check whether Master1 is active management node of MRS Manager. For details about the command, see **Step 2.4**.
2. Log in to the Master1 node using the password as user **root**.
3. Run the following commands to switch to user **omm**:

   sudo su - root

   su - omm
4. Run the following command to check the active management node of MRS Manager:

   sh ${BIGDATA_HOME}/om-0.0.1/sbin/status-oms.sh

   In the command output, the node whose **HAActive** is **active** is the active management node, and the node whose **HAActive** is **standby** is the standby management node. In the following example, **mgtomsdat-sh-3-01-1** is the active management node, and **mgtomsdat-sh-3-01-2** is the standby management node.

```
Ha mode
double
NodeName            HostName                HAVersion       StartTime           HAActive
HAAllResOK          HARunPhase
192-168-0-30        mgtomsdat-sh-3-01-1     V100R001C01     2021-11-18 23:43:02
active              normal                  Actived
192-168-0-24        mgtomsdat-sh-3-01-2     V100R001C01     2021-11-21 07:14:02
standby             normal                  Deactived
```

5. Log in to the active management node, for example, **192-168-0-30** of MRS Manager as user **root**, and run the following command to switch to user **omm**:

   sudo su - omm

Step 3 Run the following command to switch to the client installation directory, for example, **/opt/client**:

cd /opt/client

Step 4 Run the following command to update the client configuration for the active management node.

sh refreshConfig.sh /opt/client *Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/client /tmp/MRS-client/MRS_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully.

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

Step 5 Use the client on a Master node.

1. On the active management node where the client is updated, for example, node **192-168-0-30**, run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *MRS cluster user*

   For example, **kinit hbaseuser**.

4. Run the following HBase client command:

   hbase shell

Step 6 Run the following commands on the HBase client to implement service A.

1. Create the **user_info** user information table according to **Table 12-162** and add data to it.

   create '*user_info*',{**NAME** => 'i'}

   For example, to add information about the user whose ID is 12005000201, run the following commands:

put '*user_info*','*12005000201*','***i:name***','*A*'

put '*user_info*','*12005000201*','***i:gender***','*Male*'

put '*user_info*','*12005000201*','***i:age***','*19*'

put '*user_info*','*12005000201*','***i:address***','*City A*'

2. Add users' educational backgrounds and titles to the **user_info** table.

   For example, to add educational background and title information about user 12005000201, run the following commands:

   put '*user_info*','*12005000201*','**i:degree**','*master*'

   put '*user_info*','*12005000201*','**i:pose**','*manager*'

3. Query user names and addresses by user ID.

   For example, to query the name and address of user 12005000201, run the following command:

   scan'*user_info*',
   {**STARTROW**=>'*12005000201*',**STOPROW**=>'*12005000201*',**COLUMNS**=>['**i:name**','**i:address**']}

4. Query information by user name.

   For example, to query information about user A, run the following command:

   scan'*user_info*',{**FILTER**=>"SingleColumnValueFilter('i','name',=,'binary:*A*')"}

5. Delete user data from the user information table.

   All user data needs to be deleted. For example, to delete data of user 12005000201, run the following command:

   delete'*user_info*','*12005000201*','i'

6. Delete the user information table.

   disable'*user_info*'

   drop '*user_info*'

----End

For MRS 3.x or later, perform the following operations:

Step 1 Use the client on the active management node.

1. Log in to the node where the client is installed as the client installation user and run the following command to switch to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables.. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *MRS cluster user*

   For example, **kinit hbaseuser**.

4. Run the following HBase client command:

   hbase shell

Step 2 Run the following commands on the HBase client to implement service A.

1. Create the **user_info** user information table according to **Table 12-162** and add data to it.

   create '*user_info*',{**NAME** => 'i'}

   For example, to add information about the user whose ID is **12005000201**, run the following commands:

   put '*user_info*','*12005000201*','***i:name***','*A*'

   put '*user_info*','*12005000201*','***i:gender***','*Male*'

   put '*user_info*','*12005000201*','***i:age***','*19*'

   put '*user_info*','*12005000201*','***i:address***','*City A*'

2. Add users' educational backgrounds and titles to the **user_info** table.

   For example, to add educational background and title information about user 12005000201, run the following commands:

   put '*user_info*','*12005000201*','**i:degree**','*master*'

   put '*user_info*','*12005000201*','**i:pose**','*manager*'

3. Query user names and addresses by user ID.

   For example, to query the name and address of user 12005000201, run the following command:

   scan'*user_info*',
   {**STARTROW**=>'*12005000201*',**STOPROW**=>'*12005000201*',**COLUMNS**=>['**i:name**','**i:address**']}

4. Query information by user name.

   For example, to query information about user A, run the following command:

   scan'*user_info*',{**FILTER**=>"SingleColumnValueFilter('i','name',=,'binary:*A*')"}

5. Delete user data from the user information table.

   All user data needs to be deleted. For example, to delete data of user 12005000201, run the following command:

   delete'*user_info*','*12005000201*','i'

6. Delete the user information table.

   disable'*user_info*'

   drop '*user_info*'

----End

12.8.2 Using an HBase Client

Scenario

This section describes how to use the HBase client in an O&M scenario or a service scenario.

Prerequisites

- The client has been installed. For example, the installation directory is **/opt/hadoopclient**. The client directory in the following operations is only an example. Change it to the actual installation directory.

- Service component users are created by the administrator as required.

  A machine-machine user needs to download the **keytab** file and a human-machine user needs to change the password upon the first login.
- If a non-**root** user uses the HBase client, ensure that the owner of the HBase client directory is this user. Otherwise, run the following command to change the owner.

  chown user:group -R *Client installation directory*/**HBase**

Using the HBase Client (Versions Earlier Than MRS 3.x)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables.. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Component service user*

For example, **kinit hbaseuser**.

Step 5 Run the following HBase client command:

hbase shell

----End

Using the HBase Client (MRS 3.x or Later)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If you use the client to connect to a specific HBase instance in a scenario where multiple HBase instances are installed, run the following command to load the environment variables of the instance. Otherwise, skip this step. For example, to load the environment variables of the HBase2 instance, run the following command:

source HBase2/component_env

Step 5 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables.. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Component service user*

For example, **kinit hbaseuser**.

Step 6 Run the following HBase client command:

hbase shell

----End

Common HBase client commands

The following table lists common HBase client commands. For more commands, see **http://hbase.apache.org/2.2/book.html**.

Table 12-163 HBase client commands

Command	Description
create	Used to create a table, for example, **create 'test', 'f1', 'f2', 'f3'**.
disable	Used to disable a specified table, for example, **disable 'test'**.
enable	Used to enable a specified table, for example, **enable 'test'**.
alter	Used to alter the table structure. You can run the **alter** command to add, modify, or delete column family information and table-related parameter values, for example, **alter 'test', {NAME => 'f3', METHOD => 'delete'}**.
describe	Used to obtain the table description, for example, **describe 'test'**.
drop	Used to delete a specified table, for example, **drop 'test'**. Before deleting a table, you must stop it.
put	Used to write the value of a specified cell, for example, **put 'test','r1','f1:c1','myvalue1'**. The cell location is unique and determined by the table, row, and column.
get	Used to get the value of a row or the value of a specified cell in a row, for example, **get 'test','r1'**.
scan	Used to query table data, for example, **scan 'test'**. The table name and scanner must be specified in the command.

12.8.3 Creating HBase Roles

Scenario

This section guides the system administrator to create and configure an HBase role on Manager. The HBase role can set HBase administrator permissions and read (R), write (W), create (C), execute (X), or manage (A) permissions for HBase tables and column families.

Users can create a table, query/delete/insert/update data, and authorize others to access HBase tables after they set the corresponding permissions for the specified databases or tables on HDFS.

NOTE

- This section applies to MRS 3.*x* or later.
- HBase roles can be created in security mode, but cannot be created in normal mode.
- If the current component uses Ranger for permission control, you need to configure related policies based on Ranger for permission management. For details, see **Adding a Ranger Access Permission Policy for HBase**.

Prerequisites

- The system administrator has understood the service requirements.
- You have logged in to Manager.

Procedure

Step 1 On Manager, choose **System** > **Permission** > **Role**.

Step 2 On the displayed page, click **Create Role** and enter a **Role Name** and **Description**.

Step 3 Set **Permission**. For details, see **Table 12-164**.

HBase permissions:

- HBase Scope: Authorizes HBase tables. The minimum permission is read (R) and write (W) for columns.
- HBase administrator permission: HBase administrator permissions.

NOTE

Users have the read (R), write (W), create (C), execute (X), and administrate (A) permissions for the tables created by themselves.

Table 12-164 Setting a role

Task	Role Authorization
Setting the HBase administrator permission	In **Configure Resource Permission**, choose *Name of the desired cluster* > **HBase** and select **HBase Administrator Permission**.
Setting the permission for users to create tables	1. In **Configure Resource Permission**, choose *Name of the desired cluster* > **HBase** > **HBase Scope**. 2. Click **global**. 3. In the **Permission** column of the specified namespace, select **Create** and **Execute**. For example, select **Create** and **Execute** for the default namespace **default**.

Task	Role Authorization
Setting the permission for users to write data to tables	1. In **Configure Resource Permission**, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global**. 2. In the **Permission** column of the specified namespace, select **Write**. For example, select **Write** for the default namespace **default**. By default, HBase sub-objects inherit the permission from the parent object.
Setting the permission for users to read data from tables	1. In **Configure Resource Permission**, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global**. 2. In the **Permission** column of the specified namespace, select **Read**. For example, select **Read** for the default namespace **default**. By default, HBase sub-objects inherit the permission from the parent object.
Setting the permission for users to manage namespaces or tables	1. In **Configure Resource Permission**, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global**. 2. In the **Permission** column of the specified namespace, select **Manage**. For example, select **Manage** for the default namespace **default**.

Task	Role Authorization
Setting the permission for reading data from or writing data to columns	1. In **Configure Resource Permission**, select *Name of the desired cluster* > **HBase** > **HBase Scope** > **global** and click the specified namespace to display the tables in the namespace. 2. Click a table. 3. Click a column family. 4. Confirm whether you want to create a role? – If yes, enter the column name in the **Resource Name** text box. Use commas (,) to separate multiple columns. Select **Read** or **Write**. If there are no columns with the same name in the HBase table, a newly created column with the same name as the existing column has the same permission as the existing one. The column permission is set successfully. – If no, modify the column permission of the existing HBase role. The columns for which the permission has been separately set are displayed in the table. Go to **Step 3.5**. 5. To add column permissions for a role, enter the column name in the **Resource Name** text box and set the column permissions. To modify column permissions for a role, enter the column name in the **Resource Name** text box and set the column permissions. Alternatively, you can directly modify the column permissions in the table. If the column permissions are modified in the table and column permissions with the same name are added, the settings cannot be saved. You are advised to modify the column permission of a role directly in the table. The search function is supported.

Step 4 Click **OK**, and return to the **Role** page.

----End

12.8.4 Configuring HBase Replication

Scenario

As a key feature to ensure high availability of the HBase cluster system, HBase cluster replication provides HBase with remote data replication in real time. It provides basic O&M tools, including tools for maintaining and re-establishing active/standby relationships, verifying data, and querying data synchronization progress. To achieve real-time data replication, you can replicate data from the HBase cluster to another one.

Prerequisites

- The active and standby clusters have been successfully installed and started (the cluster status is **Running** on the **Active Clusters** page), and you have the administrator rights of the clusters.
- The network between the active and standby clusters is normal and ports can be used properly.
- Cross-cluster mutual trust has been configured.
- If historical data exists in the active cluster and needs to be synchronized to the standby cluster, cross-cluster replication must be configured for the active and standby clusters. For details, see **Enabling Cross-Cluster Copy**.
- Time is consistent between the active and standby clusters and the Network Time Protocol (NTP) service on the active and standby clusters uses the same time source.
- Mapping relationships between the names of all hosts in the active and standby clusters and service IP addresses have been configured in the **/etc/hosts** file by appending **192.***.***.*** **host1** to the **hosts** file.
- The network bandwidth between the active and standby clusters is determined based on service volume, which cannot be less than the possible maximum service volume.

Constraints

- Despite that HBase cluster replication provides the real-time data replication function, the data synchronization progress is determined by several factors, such as the service loads in the active cluster and the health status of processes in the standby cluster. In normal cases, the standby cluster should not take over services. In extreme cases, system maintenance personnel and other decision makers determine whether the standby cluster takes over services according to the current data synchronization indicators.
- Currently, the replication function supports only one active cluster and one standby cluster in HBase.
- Typically, do not perform operations on data synchronization tables in the standby cluster, such as modifying table properties or deleting tables. If any misoperation on the standby cluster occurs, data synchronization between the active and standby clusters will fail and data of the corresponding table in the standby cluster will be lost.
- If the replication function of HBase tables in the active cluster is enabled for data synchronization, after modifying the structure of a table in the active cluster, you need to manually modify the structure of the corresponding table in the standby cluster to ensure table structure consistency.

Procedure

Enable the replication function for the active cluster to synchronize data written by Put.

Step 1 Log in to the MRS console, click a cluster name and choose **Components**.

Step 2 Go to the **All Configurations** page of the HBase service. For details, see **Modifying Cluster Service Configuration Parameters**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 3 (Optional) Set configuration items listed in **Table 12-165**. You can set the parameters based on the description or use the default values.

Table 12-165 Optional configuration items

Navigation Path	Parameter	Default Value	Description
HMaster > Performance	hbase.master.logcleaner.ttl	600000	Time to live (TTL) of HLog files. If the value is set to **604800000** (unit: millisecond), the retention period of HLog is 7 days.
	hbase.master.cleaner.interval	60000	Interval for the HMaster to delete historical HLog files. The HLog that exceeds the configured period will be automatically deleted. You are advised to set it to the maximum value to save more HLogs.
RegionServer > Replication	replication.source.size.capacity	16777216	Maximum size of edits, in bytes. If the edit size exceeds the value, HLog edits will be sent to the standby cluster.
	replication.source.nb.capacity	25000	Maximum number of edits, which is another condition for triggering HLog edits to be sent to the standby cluster. After data in the active cluster is synchronized to the standby cluster, the active cluster reads and sends data in HLog according to this parameter value. This parameter is used together with **replication.source.size.capacity**.
	replication.source.maxretriesmultiplier	10	Maximum number of retries when an exception occurs during replication.
	replication.source.sleepforretries	1000	Retry interval (unit: ms)

Navigation Path	Parameter	Default Value	Description
	hbase.regionserver.replication.handler.count	6	Number of replication RPC server instances on RegionServer

Enable the replication function for the active cluster to synchronize data written by bulkload.

Step 4 Determine whether to enable bulkload replication.

NOTE

If bulkload import is used and data needs to be synchronized, you need to enable Bulkload replication.

If yes, go to **Step 5**.

If no, go to **Step 9**.

Step 5 Go to the **All Configurations** page of the HBase service parameters by referring to **Modifying Cluster Service Configuration Parameters**.

Step 6 On the HBase configuration interface of the active and standby clusters, search for **hbase.replication.cluster.id** and modify it. It specifies the HBase ID of the active and standby clusters. For example, the HBase ID of the active cluster is set to **replication1** and the HBase ID of the standby cluster is set to **replication2** for connecting the active cluster to the standby cluster. To save data overhead, the parameter value length is not recommended to exceed 30.

Step 7 On the HBase configuration interface of the standby cluster, search for **hbase.replication.conf.dir** and modify it. It specifies the HBase configurations of the active cluster client used by the standby cluster and is used for data replication when the bulkload data replication function is enabled. The parameter value is a path name, for example, **/home**.

NOTE

- In versions earlier than MRS 3.x, you do not need to set this parameter. Skip **Step 7**.
- When bulkload replication is enabled, you need to manually place the HBase client configuration files (**core-site.xml**, **hdfs-site.xml**, and **hbase-site.xml**) in the active cluster on all RegionServer nodes in the standby cluster. The actual path for placing the configuration file is **${hbase.replication.conf.dir}/${hbase.replication.cluster.id}**. For example, if **hbase.replication.conf.dir** of the standby cluster is set to **/home** and **hbase.replication.cluster.id** of the active cluster is set to **replication1**, the actual path for placing the configuration files in the standby cluster is **/home/replication1**. You also need to change the corresponding directory and file permissions by running the **chown -R omm:wheel /home/replication1** command.
- You can obtain the client configuration files from the client in the active cluster, for example, the **/opt/client/HBase/hbase/conf** path.

Step 8 On the HBase configuration page of the active cluster, search for and change the value of **hbase.replication.bulkload.enabled** to **true** to enable bulkload replication.

Restarting the HBase service and install the client

Step 9 Save the configurations and restart HBase.

Step 10 In the active and standby clusters, update the client configuration file.

Synchronize table data of the active cluster. (Skip this step if the active cluster has no data.)

Step 11 Access the HBase shell of the active cluster as user **hbase**.

1. On the active management node where the client has been updated, run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit hbase

   NOTE

   The system prompts you to enter the password after you run **kinit hbase**. The default password of user **hbase** is **Hbase@123**.

4. Run the following HBase client command:

   hbase shell

Step 12 Check whether historical data exists in the standby cluster. If historical data exists and data in the active and standby clusters must be consistent, delete data from the standby cluster first.

1. On the HBase shell of the standby cluster, run the **list** command to view the existing tables in the standby cluster.
2. Delete data tables from the standby cluster based on the output list.

   disable '*tableName*'

   drop '*tableName*'

Step 13 After HBase replication is configured and data synchronization is enabled, check whether tables and data exist in the active cluster and whether the historical data needs to be synchronized to the standby cluster.

Run the **list** command to check the existing tables in the active cluster and run the **scan** '*tableName*' command to check whether the tables contain historical data.

- If tables exist and data needs to be synchronized, go to **Step 14**.
- If no, no further action is required.

Step 14 The HBase replication configuration does not support automatic synchronization of historical data in tables. You need to back up the historical data of the active cluster and then manually synchronize the historical data to the standby cluster.

Manual synchronization refers to the synchronization of a single table that is implemented by Export, distcp, and Import.

The process for manually synchronizing data of a single table is as follows:

1. Export table data from the active cluster.

   hbase org.apache.hadoop.hbase.mapreduce.Export -Dhbase.mapreduce.include.deleted.rows=true *Table name Directory where the source data is stored*

   Example: **hbase org.apache.hadoop.hbase.mapreduce.Export -Dhbase.mapreduce.include.deleted.rows=true t1 /user/hbase/t1**

2. Copy the data that has been exported to the standby cluster.

   hadoop distcp *Directory for storing source data in the active cluster* **hdfs://***ActiveNameNodeIP*:**9820/** *Directory for storing source data in the standby cluster*

   ActiveNameNodeIP indicates the IP address of the active NameNode in the standby cluster.

   Example: **hadoop distcp /user/hbase/t1 hdfs://192.168.40.2:9820/user/hbase/t1**

3. Import data to the standby cluster as the HBase table user of the standby cluster.

   hbase org.apache.hadoop.hbase.mapreduce.Import -*Dimport.bulk.output=Directory where the output data is stored in the standby cluster Table name Directory where the source data is stored in the standby cluster*

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles *Directory where the output data is stored in the standby cluster Table name*

   For example, **hbase org.apache.hadoop.hbase.mapreduce.Import -Dimport.bulk.output=/user/hbase/output_t1 t1 /user/hbase/t1** and

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles /user/hbase/output_t1 t1

Add the replication relationship between the active and standby clusters.

Step 15 Run the following command on the HBase Shell to create the replication synchronization relationship between the active cluster and the standby cluster:

add_peer '*Standby cluster ID', CLUSTER_KEY => 'ZooKeeper address of the standby cluster*',**{HDFS_CONFS => true}**

- *Standby cluster ID* indicates an ID for the active cluster to recognize the standby cluster. It is recommended that the ID contain letters and digits.
- The ZooKeeper address of the standby cluster includes the service IP address of ZooKeeper, the port for listening to client connections, and the HBase root directory of the standby cluster on ZooKeeper.
- **{HDFS_CONFS => true}** indicates that the default HDFS configuration of the active cluster will be synchronized to the standby cluster. This parameter is used for HBase of the standby cluster to access HDFS of the active cluster. If bulkload replication is disabled, you do not need to use this parameter.

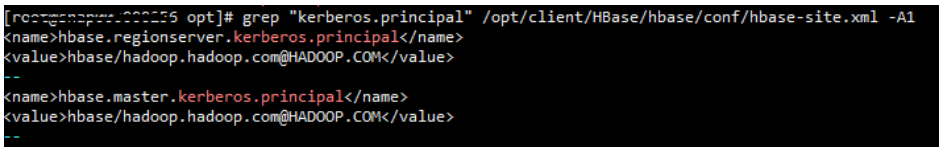
  Suppose the standby cluster ID is replication2 and the ZooKeeper address of the standby cluster is **192.168.40.2,192.168.40.3,192.168.40.4:2181:/hbase**.

  – Run the **add_peer 'replication2',CLUSTER_KEY => '192.168.40.2,192.168.40.3,192.168.40.4:2181:/hbase',CONFIG => { "hbase.regionserver.kerberos.principal" => "<val>", "hbase.master.kerberos.principal" => "<val2>" }** command for a

security cluster and the **add_peer 'replication2',CLUSTER_KEY => '192.168.40.2,192.168.40.3,192.168.40.4:2181:/hbase'** command for a common cluster.

The **hbase.master.kerberos.principal** and **hbase.regionserver.kerberos.principal** parameters are the Kerberos users of HBase in the security cluster. You can search the **hbase-site.xml** file on the client for the parameter values. For example, if the client is installed in the **/opt/client** directory of the Master node, you can run the **grep "kerberos.principal" /opt/client/HBase/hbase/conf/hbase-site.xml -A1** command to obtain the principal of HBase. See the following figure.

Figure 12-16 Obtaining the principal of HBase



NOTE

1. Obtain the ZooKeeper service IP address.

   Log in to the MRS console, click the cluster name, and choose **Components** > **ZooKeeper** > **Instances** to obtain the ZooKeeper service IP address.
2. On the ZooKeeper service parameter configuration page, search for clientPort, which is the port for the client to connect to the server.
3. Run the **list_peers** command to check whether the replication relationship between the active and standby clusters is added. If the following information is displayed, the relationship is successfully added.

```
hbase(main):003:0> list_peers
PEER_ID CLUSTER_KEY ENDPOINT_CLASSNAME STATE REPLICATE_ALL NAMESPACES
TABLE_CFS BANDWIDTH SERIAL
replication2 192.168.0.13,192.168.0.177,192.168.0.25:2181:/hbase ENABLED  true   0 false
```

Specify the data writing status for the active and standby clusters.

Step 16 On the HBase shell of the active cluster, run the following command to retain the data writing status:

set_clusterState_active

The command is run successfully if the following information is displayed:

```
hbase(main):001:0> set_clusterState_active
=> true
```

Step 17 On the HBase shell of the standby cluster, run the following command to retain the data read-only status:

set_clusterState_standby

The command is run successfully if the following information is displayed:

```
hbase(main):001:0> set_clusterState_standby
=> true
```

Enable the HBase replication function to synchronize data.

Step 18 Check whether a namespace exists in the HBase service instance of the standby cluster and the namespace has the same name as the namespace of the HBase table for which the replication function is to be enabled.

On the HBase shell of the standby cluster, run the **list_namespace** command to query the namespace.

- If the same namespace exists, go to **Step 19**.
- If the same namespace does not exist, on the HBase shell of the standby cluster, run the following command to create a namespace with the same name and go to **Step 19**:

  create_namespace'ns1

Step 19 On the HBase shell of the active cluster, run the following command to enable real-time replication for tables in the active cluster. This ensures that modified data in the active cluster can be synchronized to the standby cluster in real time.

You can only synchronize data of one HTable at one time.

enable_table_replication '*Table name***'**

NOTE

- If the standby cluster does not contain a table with the same name as the table for which real-time synchronization is to be enabled, the table is automatically created.
- If a table with the same name as the table for which real-time synchronization is to be enabled exists in the standby cluster, the structures of the two tables must be the same.
- If the encryption algorithm SMS4 or AES is configured for '*Table name*', the function for synchronizing data from the active cluster to the standby cluster cannot be enabled for the HBase table.
- If the standby cluster is offline or has tables with the same name but different structures, the replication function cannot be enabled.

  If the standby cluster is offline, start it.

  If the standby cluster has a table with the same name but different structure, modify the table structure to make it as the same as the table structure of the active cluster. On the HBase shell of the standby cluster, run the ***alter*** command to change the password by referring to the example.

Step 20 On the HBase shell of the active cluster, run the following command to enable the real-time replication function for the active cluster to synchronize the HBase permission table:

enable_table_replication 'hbase:acl'

NOTE

After the permission of the active HBase source data table is modified, to ensure that the standby cluster can properly read data, modify the role permission for the standby cluster.

Check the data synchronization status for the active and standby clusters.

Step 21 Run the following command on the HBase client to check the synchronized data of the active and standby clusters. After the replication function is enabled, you can run this command to check whether the newly synchronized data is consistent.

hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication --starttime=*Start time* **--endtime**=*End time Column family name ID of the standby cluster Table name*

NOTE

- The start time must be earlier than the end time.
- The value of **starttime** and **endtime** must be in the timestamp format. You need to run **date -d "2015-09-30 00:00:00" +%s to** change a common time format to a timestamp format. The command output is a 10-digit number (accurate to second), but HBase identifies a 13-digit number (accurate to millisecond). Therefore, you need to add three zeros (000) to the end of the command output.

Switch over active and standby clusters.

NOTE

1. If the standby cluster needs to be switched over to the active cluster, reconfigure the active/standby relationship by referring to **Step 2** to **Step 10** and **Step 15** to **Step 20**.
2. Do not perform **Step 11** to **Step 14**.

----End

Related Commands

Table 12-166 HBase replication

Operation	Command	Description
Set up the active/standby relationship.	**add_peer** *'Standby cluster ID', 'Standby cluster address'* Examples: **add_peer '1', 'zk1,zk2,zk3:2181:/hbase'** **add_peer '1', 'zk1,zk2,zk3:2181:/hbase1'**	Set up the relationship between the active cluster and the standby cluster. To enable bulkload replication, run the **add_peer** *'Standby cluster ID'*,***CLUSTER_KEY =>*** *'Standby cluster address'* command, configure **hbase.replication.conf.dir**, and manually copy the HBase client configuration file in the active cluster to all RegionServer nodes in the standby cluster. For details, see **Step 4** to **11**.
Remove the active/standby relationship.	**remove_peer** *'Standby cluster ID'* Example: **remove_peer '1'**	Remove standby cluster information from the active cluster.
Query the active/standby relationship.	**list_peers**	Query standby cluster information (mainly Zookeeper information) in the active cluster.

Operation	Command	Description
Enable the real-time user table synchronization function.	**enable_table_replication** *'Table name'* Example: **enable_table_replication 't1'**	Synchronize user tables from the active cluster to the standby cluster.
Disable the real-time user table synchronization function.	**disable_table_replication** *'Table name'* Example: **disable_table_replication 't1'**	Do not synchronize user tables from the active cluster to the standby cluster.
Verify data of the active and standby clusters.	**bin/hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication** *--starttime --endtime Column family name Standby cluster ID Table name*	Verify whether data of the specified table is the same between the active cluster and the standby cluster. The description of the parameters in this command is as follows: ● Start time: If start time is not specified, the default value **0** will be used. ● End time: If end time is not specified, the time when the current operation is submitted will be used by default. ● Table name: If a table name is not entered, all user tables for which the real-time synchronization function is enabled will be verified by default.
Switch the data writing status.	**set_clusterState_active** **set_clusterState_standby**	Specifies whether data can be written to the cluster HBase tables.

Operation	Command	Description
Add or update the active cluster HDFS configurations saved in the peer cluster.	**set_replication_hdfs_confs 'PeerId', {'key1' => 'value1', 'key2' => 'value2'}**	Enable replication for data including bulkload data. When HDFS parameters are modified in the active cluster, the modification cannot be automatically synchronized to the standby cluster. You need to manually run the command to synchronize the changes. The affected parameters are as follows: • fs.defaultFS • dfs.client.failover.proxy.provider.hacluster • dfs.client.failover.connection.retries.on.timeouts • dfs.client.failover.connection.retries For example, if the value of **fs.defaultFS** is changed to **hdfs://hacluster_sale**, run the **set_replication_hdfs_confs '1', {'fs.defaultFS' => 'hdfs://hacluster_sale'}** command to synchronization the HDFS configuration to the standby cluster whose ID is 1.

12.8.5 Configuring HBase Parameters

NOTE

The operations described in this section apply only to clusters of versions earlier than MRS 3.x.

If the default parameter settings of the MRS service cannot meet your requirements, you can modify the parameter settings as required.

Step 1 Go to the cluster details page and choose **Components**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 2 Choose **HBase** > **Service Configuration** and switch **Basic** to **All**. On the displayed HBase configuration page, modify parameter settings.

Table 12-167 HBase parameters

Parameter	Description	Value
hbase.regionserver.hfile.durable.sync	Whether to enable the HFile durability to make data persistence on disks. If this parameter is set to **true**, HBase performance is affected because each HFile is synchronized to disks by **hadoop fsync** when being written to HBase. This parameter exists only in MRS 1.9.2 or earlier.	Possible values are as follows: • **true** • **false** The default value is **true**.
hbase.regionserver.wal.durable.sync	Specifies whether to enable WAL file durability to make the WAL data persistence on disks. If this parameter is set to **true**, HBase performance is affected because each edited WAL file is synchronized to disks by **hadoop fsync** when being written to HBase. This parameter exists only in MRS 1.9.2 or earlier.	Possible values are as follows: • **true** • **false** The default value is **true**.

----End

12.8.6 Enabling Cross-Cluster Copy

Scenario

DistCp is used to copy the data stored on HDFS from a cluster to another cluster. DistCp depends on the cross-cluster copy function, which is disabled by default. This function needs to be enabled in both clusters.

This section describes how to enable cross-cluster copy.

Impact on the System

Yarn needs to be restarted to enable the cross-cluster copy function and cannot be accessed during the restart.

Prerequisites

The **hadoop.rpc.protection** parameter of the two HDFS clusters must be set to the same data transmission mode, which can be **privacy** (encryption enabled) or **authentication** (encryption disabled).

NOTE

Go to the **All Configurations** page by referring to **Modifying Cluster Service Configuration Parameters** and search for **hadoop.rpc.protection**.

For versions earlier than MRS 3.x, choose **Components** > **HDFS** > **Service Configuration** on the cluster details page. Switch **Basic** to **All**, and search for **hadoop.rpc.protection**.

Procedure

Step 1 Go to the **All Configurations** page of the Yarn service. For details, see **Modifying Cluster Service Configuration Parameters**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 2 In the navigation pane, choose **Yarn** > **Distcp**.

Step 3 Set **haclusterX.remotenn1** of **dfs.namenode.rpc-address** to the service IP address and RPC port number of one NameNode instance of the peer cluster, and set **haclusterX.remotenn2** to the service IP address and RPC port number of the other NameNode instance of the peer cluster. Enter a value in the *IP address:port* format.

NOTE

You can log in to FusionInsight Manager in MRS 3.x clusters, and choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Instance** to obtain the service IP address of the NameNode instance.

For versions earlier than MRS 3.x, on the cluster details page, choose **Components** > **HDFS** > **Instances** to obtain the service IP address of the NameNode instance.

dfs.namenode.rpc-address.haclusterX.remotenn1 and **dfs.namenode.rpc-address.haclusterX.remotenn2** do not distinguish active and standby NameNode instances. The default NameNode RPC port is 9820 and cannot be modified on MRS Manager.

For example, **10.1.1.1:9820** and **10.1.1.2:9820**.

Step 4 Save the configuration. On the **Dashboard** tab page, and choose **More** > **Restart Service** to restart the Yarn service.

Operation succeeded is displayed. Click **Finish**. The Yarn service is started successfully.

Step 5 Log in to the other cluster and repeat the preceding operations.

----End

12.8.7 Using the ReplicationSyncUp Tool

Prerequisites

1. Active and standby clusters have been installed and started.
2. Time is consistent between the active and standby clusters and the NTP service on the active and standby clusters uses the same time source.

3. When the HBase service of the active cluster is stopped, the ZooKeeper and HDFS services must be started and run.
4. ReplicationSyncUp must be run by the system user who starts the HBase process.
5. In security mode, ensure that the HBase system user of the standby cluster has the read permission on HDFS of the active cluster. This is because that it will update the ZooKeeper nodes and HDFS files of the HBase system.
6. When HBase of the active cluster is faulty, the ZooKeeper, file system, and network of the active cluster are still available.

Scenarios

The replication mechanism can use WAL to synchronize the state of a cluster with the state of another cluster. After HBase replication is enabled, if the active cluster is faulty, ReplicationSyncUp synchronizes incremental data from the active cluster to the standby cluster using the information from the ZooKeeper node. After data synchronization is complete, the standby cluster can be used as an active cluster.

Parameter Configuration

Parameter	Description	Default Value
hbase.replication.bulkload.enabled	Whether to enable the bulkload data replication function. The parameter value type is Boolean. To enable the bulkload data replication function, set this parameter to **true** for the active cluster.	**false**
hbase.replication.cluster.id	ID of the source HBase cluster. After the bulkload data replication is enabled, this parameter is mandatory and must be defined in the source cluster. The parameter value type is String.	-

Tool Usage

Run the following command on the client of the active cluster:

hbase org.apache.hadoop.hbase.replication.regionserver.ReplicationSyncUp -Dreplication.sleep.before.failover=1

NOTE

replication.sleep.before.failover indicates sleep time required for replication of the remaining data when RegionServer fails to start. You are advised to set this parameter to 1 second to quickly trigger replication.

Precautions

1. When the active cluster is stopped, this tool obtains the WAL processing progress and WAL processing queue from the ZooKeeper Node (RS znode) and copies the queues that are not copied to the standby cluster.
2. RegionServer of each active cluster has its own znode under the replication node of ZooKeeper in the standby cluster. It contains one znode of each peer cluster.
3. If RegionServer is faulty, each RegionServer in the active cluster receives a notification through the watcher and attempts to lock the znode of the faulty RegionServer, including its queues. The successfully created RegionServer transfers all queues to the znode of its own queue. After queues are transferred, they are deleted from the old location.
4. When the active cluster is stopped, ReplicationSyncUp synchronizes data between active and standby clusters using the information from the ZooKeeper node. In addition, WALs of the RegionServer znode will be moved to the standby cluster.

Restrictions and Limitations

If the standby cluster is stopped or the peer relationship is closed, the tool runs normally but the peer relationship cannot be replicated.

12.8.8 Using HIndex

12.8.8.1 Introduction to HIndex

Scenarios

HBase is a distributed storage database of the Key-Value type. Data in tables is sorted by dictionary based on row keys. If you query data by specifying a row key or scan data in a specific row key range, HBase can help you quickly locate the data to be read. In most cases, you need to query data whose column value is *XXX*. HBase provides the filter function to enable you to query data with a specific column value. All data is scanned in the sequence of row keys and is matched with the specific column value until the required data is found. To obtain the required data, the filter will scan some unnecessary data. As a result, the filter function cannot meet the requirements for high-performance, frequent queries.

HBase HIndex is designed to address these issues. HBase HIndex provides HBase with the capability of indexing based on specific column values, making queries faster.

NOTE

- Rolling upgrade is not supported for index data.
- Composite index: You must add or delete all columns that participate in composite indexes. Otherwise, the data may be inconsistent.
- You should not explicitly configure any split policy to a data table where an index has been created.
- The mutation operations are not supported, such as increment and append.
- Index of the column with **maxVersions** greater than 1 is not supported.
- The value size of a column for which an index is added cannot exceed 32 KB.
- When the user data is deleted because TTL of the column family is invalid, the corresponding index data will not be deleted immediately. The index data will be deleted during major compaction.
- After an index is created, the TTL of the user column family must not be changed.
  - If the TTL of the column family is changed to a larger value after an index is created, delete the index and create one again. Otherwise, some generated index data may be deleted before the deletion of user data.
  - If the TTL of the column family is changed to a smaller value after an index is created, the index may be deleted after the deletion of user data.
- After disaster recovery is enabled for HBase tables, a secondary index is created in the active cluster and index table changes are not automatically synchronized to the standby cluster. To implement disaster recovery in this case, perform the following operations:
  1. After the secondary index is created in the active table, create a secondary index with the same schema and name using the same method in the standby cluster.
  2. In the active cluster, manually set **REPLICATION_SCOPE** of the index column family (default value: **d**) to **1**.

Parameter Configuration

1. Log in to the MRS console, click a cluster name and choose **Components**.
2. Go to the **All Configurations** page of the HBase service. For details, see **Modifying Cluster Service Configuration Parameters**.
3. View parameters on the HBase configurations page.

Navigation Path	Parameter	Default Value	Description
HMaster > System	hbase.coprocessor.master.classes	org.apache.hadoop.hbase.hindex.server.master.HIndexMasterCoprocessor,com.xxx.hadoop.hbase.backup.services.RecoveryCoprocessor,org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor,org.apache.hadoop.hbase.security.access.ReadOnlyClusterEnabler,org.apache.hadoop.hbase.rsgroup.RSGroupAdminEndpoint	This coprocessor is used to handle Master-level operations after the HIndex function is enabled, for example, creating an index meta table, adding an index, and deleting an index, a table, and index metadata.
RegionServer > RegionServer	hbase.coprocessor.regionserver.classes	org.apache.hadoop.hbase.hindex.server.regionserver.HIndexRegionServerCoprocessor,org.apache.hadoop.hbase.JMXListener,org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor	This coprocessor is used to handle the operations that the Master delivers to RegionServer after the HIndex function is enabled.

Navigation Path	Parameter	Default Value	Description
	hbase.coprocessor.region.classes	org.apache.hadoop.hbase.hindex.server.regionserver.HIndexRegionCoprocessor,org.apache.hadoop.hbase.security.token.TokenProvider,com.xxx.hadoop.hbase.backup.services.RecoveryCoprocessor,org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor,org.apache.hadoop.hbase.security.access.SecureBulkLoadEndpoint,org.apache.hadoop.hbase.security.access.ReadOnlyClusterEnabler,org.apache.hadoop.hbase.coprocessor.MetaTableMetrics	This coprocessor is used to operate data in the Region after the HIndex function is enabled.

Navigation Path	Parameter	Default Value	Description
	hbase.coprocessor.wal.classes	org.apache.hadoop.hbase.hindex.server.regionserver.HIndexRegionServerCoprocessor,org.apache.hadoop.hbase.JMXListener,org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor	This coprocessor is used for Replication, which filters index data to prevent the index data from being sent to the peer cluster. The peer cluster generates index data by itself. This parameter is supported only in versions earlier than MRS 3.x.

NOTE

1. The preceding default values need to be configured after the HBase HIndex function is enabled. In MRS clusters that support the HBase HIndex function, the values have been configured by default.

2. Ensure that the **master** parameter is configured on HMaster and the **region** and **regionserver** parameters are configured on RegionServer.

Related Interfaces

The APIs that use HIndex are in the **org.apache.hadoop.hbase.hindex.client.HIndexAdmin** class. The following table describes the related APIs.

Operation	API	Description	Precautions
Add an index.	addIndices()	Add an index to a table without data. Calling this API will add the specified index to a table but skips index data generation. Therefore, after this operation, the index cannot be used for the scanning and filtering operations. This API applies to scenarios where users want to add indexes in batches to tables that have a large amount of pre-existing user data. The specific operation is to use external tools such as the TableIndexer tool to build index data.	• An index cannot be modified once it is added. To modify the index, you need to delete the old index and then create a new one. • Do not create two indexes on the same column with different index names. Otherwise, storage and processing resources will be wasted. • Indexes cannot be added to a system table. • The append and increment operations are not supported when data is put into the index column. • If any fault occurs on the client except **DoNotRetryIOException**, you need to try again. • An index column family is selected from the following conditions in sequence based on availability: – Typically, the default index column family is **d**. However, if the value of **hindex.default.family.name** is set, the value will be used. – Symbol #, @, $, or %
	addIndicesWithData()	Add an index to a table with data. This API is used to add the specified index to the table and create index data for the existing user data. Alternatively, the API can be called to generate an index and then generate index data when the user data is being stored. Therefore, after this operation, the index can be used for the scanning and filtering operations immediately.	

Operation	API	Description	Precautions
			– #0, @ 0, $ 0, %0, #1, @ 1 ...to #255, @ 255, $ 255, %255 – Throw exceptions. ● You can use the HIndex TableIndexer tool to add indexes without building index data.
Delete an index.	dropIndices()	This API is used to delete an index only. It deletes the specified index from a table but skips the corresponding index data. After this operation, the index cannot be used for the scanning and filtering operations. The cluster automatically deletes old index data during major compaction. This API applies to scenarios where a table contains a large amount of index data and **dropIndicesWithData()** is unavailable. In addition, you can use the TableIndexer tool to delete indexes and index data.	● An index can be disabled when it is in the **ACTIVE**, **INACTIVE**, or **DROPPING** state. ● If you use **dropIndices()** to delete an index, ensure that the index data has been deleted before the index is added to the table with the same index name (that is, major compaction has been completed). ● If you delete an index, the following information will also be deleted: – A column family with an index – Any one of column families in a combination index ● Indexes and index data can be deleted together using the HIndex TableIndexer tool.
	dropIndicesWithData()	Delete index data. This API deletes the specified index and all index data corresponding to the index in a user table. After this operation, the index is completely deleted from the table and is no longer used for the scanning and filtering operations.	

Operation	API	Description	Precautions
Enable/ Disable an index.	disableIndices()	This API disables all indexes specified by a user so that they are no longer used for the scanning and filtering operations.	• An index can be enabled when the index is in the **ACTIVE**, **INACTIVE**, or **BUILDING** state. • An index can be disabled when the index is in the **ACTIVE** or **INACTIVE** state. • Before disabling an index, ensure that the index data is consistent with the user data. If no new data is added to the table when the index is disabled, the index data is consistent with the user data. • When enabling an index, you can use the TableIndexer tool to build index data to ensure data consistency.
	enableIndices()	This API enables all indexes specified by a user so that they can be used for the scanning and filtering operations.	
View the created index.	listIndices()	This API is used to list all indexes of a specified table.	N/A

Querying Data Based on Indexes

You can use a filter to query data in a user table with an index. The query result of a user table with a single or combination index is the same as that of a table without an index, but the table with an index provides higher data query performance than the table without an index.

The index usage rules are as follows:

- Scenario 1: A single index is created for one or more columns.
  - When this column is used for AND or OR query filtering, an index can improve query performance.

    Example: Filter_Condition(IndexCol1)AND / OR Filter_Condition(IndexCol2)
  - When you use **Index Column AND Non-Index Column** for filtering in the query, the index can improve query performance.

Example: Filter_Condition(IndexCol1)AND Filter_Condition(IndexCol2)AND Filter_Condition(NonIndexCol1)

- When you use **Index Column OR Non-Index Column** for filtering in the query but do not use an index, query performance will not be improved.

  Example: Filter_Condition(IndexCol1)AND / OR Filter_Condition(IndexCol2) OR Filter_Condition(NonIndexCol1)

- Scenario 2: A combination index is created for multiple columns.
  - When the columns to be queried are all or part of the combination index and have the same order as the combination index, using the index improves query performance.

    For example, create a combination index for C1, C2, and C3.

    - The index takes effect in the following situations:

      Filter_Condition(IndexCol1)AND Filter_Condition(IndexCol2)AND Filter_Condition(IndexCol3)

      Filter_Condition(IndexCol1)AND Filter_Condition(IndexCol2)

      FILTER_CONDITION(IndexCol1)

    - The index does not take effect in the following situations:

      Filter_Condition(IndexCol2)AND Filter_Condition(IndexCol3)

      Filter_Condition(IndexCol1)AND Filter_Condition(IndexCol3)

      FILTER_CONDITION(IndexCol2)

      FILTER_CONDITION(IndexCol3)

  - When you use **Index Column AND Non-Index Column** for filtering in the query, the index can improve query performance.

    Examples:

    Filter_Condition(IndexCol1)AND Filter_Condition(NonIndexCol1)

    Filter_Condition(IndexCol1)AND Filter_Condition(IndexCol2)AND Filter_Condition(NonIndexCol1)

  - When you use **Index Column OR Non-Index Column** for filtering in the query but do not use an index, query performance will not be improved.

    Examples:

    Filter_Condition(IndexCol1)OR Filter_Condition(NonIndexCol1)

    (Filter_Condition(IndexCol1)AND Filter_Condition(IndexCol2))OR(Filter_Condition(NonIndexCol1))

  - When multiple columns are used for query, you can specify a value range for only the last column in the combination index and set other columns to specified values

    For example, create a combination index for C1, C2, and C3. In a range query, only the value range of C3 can be set. The filter criteria are "C1 = *XXX*, C2 = *XXX*, and C3 = Value range."

Query Policy Selection

Use **SingleColumnValueFilter** or **SingleColumnRangeFilter**. It will provide the definite value **column_family:qualifierpair** (called **col1**) in filter criteria.

If **col1** is the first index column in the table, any index in the table can be a candidate index used during the query. The following provides an example:

If there is an index on **col1**, the index can be used as a candidate index because **col1** is the first and the only column of the index. If there is another index on **col1** and **col2**, you can consider this index as a candidate index because **col1** is the first column in the index list. However, if there is an index on **col2** and **col1**, this index cannot be used as a candidate index because the first column in the index list is not **col1**.

The most suitable method to use the index now is that when there are multiple candidate indexes, select the most suitable index for scanning data.

You can use the following solutions to learn how to select the best index policy.

- It is better to fully match.

  Scenario: There are two indexes available, one for **col1&col2** and the other for **col1**.

  In this scenario, the second index is better than the first one, because it scans less index data.

- If there are multiple candidate multi-column indexes, select an index with fewer index columns.

  Scenario: There are two indexes available, one for **col1&col2** and the other for **col1&col2&col3**.

  In this case, you had better use the index on **col1&col2**, because it scans less index data.

NOTE

- During a query based on an index, the index state must be **ACTIVE**. You can call the **listIndices()** API to view the index state.
- To query the correct data based on the index, ensure the consistency between index data and user data.
- Run the following command to perform a complex query on the HBase shell client (assuming that an index has been created for the specified column):

  scan 'tablename', {FILTER => "SingleColumnValueFilter(family, qualifier, compareOp, comparator, filterIfMissing, latestVersionOnly)"}

  Example: **scan 'test', {FILTER => "SingleColumnValueFilter('info', 'age', =, 'binary:26', true, true)"}**

  In the preceding scenario, if you want to save the row where no column is found in the result, you should not create any index in any such column, because if the column to be queried does not exist, the row will be filtered out when SCVF is used to scan the index columns. When the SCVF whose **filterIfMissingset** is **false** (default value) scans non-index columns, rows where no column is queried will also be returned in the result. Therefore, to avoid inconsistent query results, you are advised to set **filterIfMissing** to **true** after creating SCVF for the index column.
- Run the following command on the HBase shell client to view the index data created for user data:

  scan 'tablename', {ATTRIBUTES => {'FETCH_INDEX_DATA' => 'true'}}

12.8.8.2 Loading Index Data in Batches

Scenarios

HBase provides the ImportTsv&LoadIncremental tool to load user data in batches. HBase also provides the HIndexImportTsv tool to load both the user data and index data in batches. HIndexImportTsv inherits all functions of the HBase batch data loading tool ImportTsv. If a table is not created before the HIndexImportTsv tool is executed, an index will be created when the table is created, and index data is generated when user data is generated.

Procedure

1. Run the following commands to import data to HDFS:

   hdfs dfs -mkdir <*inputdir*>

   hdfs dfs -put <local_data_file> <*inputdir*>

   For example, define the data file **data.txt** as follows:

   ```
   12005000201,Zhang San,Male,19,A City, A Province
   12005000202,Li Wanting,Female,23,B City, B Province
   12005000203,Wang Ming,Male,26,C City, C Province
   12005000204,Li Gang,Male,18,D City, D Province
   12005000205,Zhao Enru,Female,21,E City, E Province
   12005000206,Chen Long,Male,32,F City, F Province
   12005000207,Zhou Wei,Female,29,G City, G Province
   12005000208,Yang Yiwen,Female,30,H City, H Province
   12005000209,Xu Bing,Male,26,I City, I Province
   12005000210,Xiao Kai,Male,25,J City, J Province
   ```

   Run the following commands:

   hdfs dfs -mkdir /datadirImport

   hdfs dfs -put data.txt /datadirImport

2. Go to HBase shell and run the following command to create the **bulkTable** table:

   create 'bulkTable', {NAME => 'info',COMPRESSION => 'SNAPPY', DATA_BLOCK_ENCODING => 'FAST_DIFF'},{NAME=>'address'}

   After the execution is complete, exit the HBase shell.

3. Run the following commands to generate an HFile file (StoreFiles):

   hbase org.apache.hadoop.hbase.hindex.mapreduce.HIndexImportTsv -Dimporttsv.separator=*<separator>*

   -Dimporttsv.bulk.output=*</path/for/output>* **-Dindexspecs.to.add=**<indexspecs> **-Dimporttsv.columns=**<columns> tableName *<inputdir>*

   - **-Dimport.separator**: indicates a separator, for example, **-Dimport.separator=','**.
   - **-Dimport.bulk.output=</path/for/output>**: indicates the output path of the execution result. You need to specify a path that does not exist.
   - **<columns>**: Indicates the mapping of the imported data in a table, for example, **-Dimporttsv.columns=HBASE_ROW_KEY,info:name,info:gender,info:age,address:city,address:province**.
   - **<tablename>**: Indicates the name of a table to be operated.

- **<inputdir>**: Indicates the directory where data is loaded in batches.
- **-Dindexspecs.to.add=<indexspecs>**: Indicates the mapping between an index name and a column, for example, **-Dindexspecs.to.add='index_bulk=>info:[age->String]'**. The index composition can be represented as follows:

  indexNameN=>familyN :[columnQualifierN-> columnQualifierDataType], [columnQualifierM-> columnQualifierDataType];familyM: [columnQualifierO-> columnQualifierDataType]# indexNameN=> familyM: [columnQualifierO-> columnQualifierDataType]

  Column qualifiers are separated by commas (,).

  Example: "index1 => f1:[c1-> String],[c2-> String]"

  Column families are separated by semicolons (;).

  Example: "index1 => f1:[c1-> String],[c2-> String]; f2:[c3-> Long]"

  Multiple indexes are separated by pound keys (#).

  Example: "index1 => f1:[c1-> String],[c2-> String]; f2:[c3-> Long]#index2 => f2:[c3-> Long]"

  The following data types are supported by columns.

  Available data types are as follows: STRING, INTEGER, FLOAT, LONG, DOUBLE, SHORT, BYTE, CHAR

  NOTE

  Data types can also be transferred in lowercase.

For example, run the following command:

hbase org.apache.hadoop.hbase.hindex.mapreduce.HIndexImportTsv -Dimporttsv.separator=',' -Dimporttsv.bulk.output=/dataOutput -Dindexspecs.to.add='index_bulk=>info:[age->String]' -Dimporttsv.columns=HBASE_ROW_KEY,info:name,info:gender,info:age,address:city,address:province bulkTable /datadirImport/data.txt

Command output:

```
[root@shap000000406 opt]# hbase org.apache.hadoop.hbase.hindex.mapreduce.HIndexImportTsv -Dimporttsv.separator=',' -Dimporttsv.bulk.output=/dataOutput -Dindexspecs.to.add='index_bulk=>info:[age->String]' -Dimporttsv.columns=HBASE_ROW_KEY,info:name,info:gender,info:age,address:city,address:province bulkTable /datadirImport/data.txt
2018-05-08 21:29:16,059 INFO  [main] mapreduce.HFileOutputFormat2: Incremental table bulkTable output configured.
2018-05-08 21:29:16,069 INFO  [main] client.ConnectionManager$HConnectionImplementation: Closing master protocol: MasterService
2018-05-08 21:29:16,069 INFO  [main] client.ConnectionManager$HConnectionImplementation: Closing zookeeper sessionid=0x80007c2cb4fd5b4d
2018-05-08 21:29:16,072 INFO  [main] zookeeper.ZooKeeper: Session: 0x80007c2cb4fd5b4d closed
2018-05-08 21:29:16,072 INFO  [main-EventThread] zookeeper.ClientCnxn: EventThread shut down for session: 0x80007c2cb4fd5b4d
2018-05-08 21:29:16,379 INFO  [main] client.ConfiguredRMFailoverProxyProvider: Failing over to 147
2018-05-08 21:29:17,328 INFO  [main] input.FileInputFormat: Total input files to process : 1
2018-05-08 21:29:17,413 INFO  [main] mapreduce.JobSubmitter: number of splits:1
2018-05-08 21:29:17,430 INFO  [main] Configuration.deprecation: io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2018-05-08 21:29:17,687 INFO  [main] mapreduce.JobSubmitter: Submitting tokens for job: job_1525338489458_0002
2018-05-08 21:29:18,100 INFO  [main] impl.YarnClientImpl: Submitted application application_1525338489458_0002
2018-05-08 21:29:18,136 INFO  [main] mapreduce.Job: The url to track the job: http://shap000000407:8088/proxy/application_1525338489458_0002/
2018-05-08 21:29:18,136 INFO  [main] mapreduce.Job: Running job: job_1525338489458_0002
```

```
2018-05-08 21:29:28,248 INFO  [main] mapreduce.Job: Job job_1525338489458_0002 running in uber mode : false
2018-05-08 21:29:28,249 INFO  [main] mapreduce.Job:  map 0% reduce 0%
2018-05-08 21:29:38,344 INFO  [main] mapreduce.Job:  map 100% reduce 0%
2018-05-08 21:29:51,421 INFO  [main] mapreduce.Job:  map 100% reduce 100%
2018-05-08 21:29:51,428 INFO  [main] mapreduce.Job: Job job_1525338489458_0002 completed successfully
2018-05-08 21:29:51,523 INFO  [main] mapreduce.Job: Counters: 50
```

4. Run the following command to import the generated HFile to HBase:

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles *</path/for/output> <tablename>*

   For example, run the following command:

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles / dataOutput bulkTable

   Command output:

```
[root@shap000000406 opt]# hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles / dataOutput bulkTable
2018-05-08 21:30:01,398 WARN  [main] mapreduce.LoadIncrementalHFiles: Skipping non-directory hdfs://hacluster/dataOutput/_SUCCESS
2018-05-08 21:30:02,006 INFO  [LoadIncrementalHFiles-0] hfile.CacheConfig: Created cacheConfig: CacheConfig:disabled
2018-05-08 21:30:02,006 INFO  [LoadIncrementalHFiles-2] hfile.CacheConfig: Created cacheConfig: CacheConfig:disabled
2018-05-08 21:30:02,006 INFO  [LoadIncrementalHFiles-1] hfile.CacheConfig: Created cacheConfig: CacheConfig:disabled
2018-05-08 21:30:02,085 INFO  [LoadIncrementalHFiles-2] compress.CodecPool: Got brand-new decompressor [.snappy]
2018-05-08 21:30:02,120 INFO  [LoadIncrementalHFiles-0] mapreduce.LoadIncrementalHFiles: Trying to load hfile=hdfs://hacluster/dataOutput/address/042426c252f74e859858c7877b95e510 first=12005000201 last=12005000210
2018-05-08 21:30:02,120 INFO  [LoadIncrementalHFiles-2] mapreduce.LoadIncrementalHFiles: Trying to load hfile=hdfs://hacluster/dataOutput/info/f3995920ae0247a88182f637aa031c49 first=12005000201 last=12005000210
2018-05-08 21:30:02,128 INFO  [LoadIncrementalHFiles-1] mapreduce.LoadIncrementalHFiles: Trying to load hfile=hdfs://hacluster/dataOutput/d/c53b252248af42779f29442ab84f86b8 first=\x00index_bulk \x00\x00\x00\x00\x00\x00\x00\x0018\x00\x0012005000204 last=\x00index_bulk \x00\x00\x00\x00\x00\x00\x00\x0032\x00\x0012005000206
2018-05-08 21:30:02,231 INFO  [main] client.ConnectionManager$HConnectionImplementation: Closing master protocol: MasterService
2018-05-08 21:30:02,231 INFO  [main] client.ConnectionManager$HConnectionImplementation: Closing zookeeper sessionid=0x81007c2cf0f55cc5
2018-05-08 21:30:02,235 INFO  [main] zookeeper.ZooKeeper: Session: 0x81007c2cf0f55cc5 closed
2018-05-08 21:30:02,235 INFO  [main-EventThread] zookeeper.ClientCnxn: EventThread shut down for session: 0x81007c2cf0f55cc5
```

12.8.8.3 Using an Index Generation Tool

Scenarios

To quickly create indexes for user data, HBase provides the TableIndexer tool for you to create, add, and delete indexes using MapReduce functions. The application scenarios are as follows:

- You want to add an index for a specified column in a table where a large amount of data exists. However, if you use the **addIndicesWithData()** API to add an index, index data corresponding to the related user data will be generated, which is time-consuming. If you use **addIndices()** to create an index, index data corresponding to user data will not be generated. Therefore, to create index data for user data, you can use the TableIndexer tool to create an index.

- If the index data is inconsistent with the user data, the tool can be used to rebuild index data.

  If you temporarily disable the index, put new data to the disabled index column, and then directly enable the index from the disabled state, index data and user data may be inconsistent. Therefore, you must rebuild all index data before using it again.
- You can use the TableIndexer tool to completely delete a large amount of existing index data from a user table.
- For user tables that do not have indexes, this tool allows you to add and build indexes at the same time.

How to Use

- **Adding a new index to a user table**

  The command is as follows:

  hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer -Dtablename.to.index=tablename -Dindexspecs.to.add='idx_0=>cf_0:[q_0->string],[q_1];cf_1:[q_2],[q_3]#*idx_1*=>cf_1:[q_4]'

  The following parameters are required.

  - **tablename.to.index**: Indicates the name of a table for which an index is created.
  - **indexspecs.to.add**: Indicates the mapping between the index name and the column in the corresponding user table.
  - **scan.caching** (optional): Contains an integer value, indicating the number of cached rows to be transmitted to the scanner during data table scanning.

  The parameters in the preceding command are described as follows:

  - **idx_1**: Indicates an index name.
  - **cf_0**: Indicates the name of a column family.
  - **q_0**: Indicates the name of a column.
  - **string**: Indicates a data type. The parameter value can be STRING, INTEGER, FLOAT, LONG, DOUBLE, SHORT, BYTE, or CHAR.

  NOTE

  - The pound key (#) is used to separate indexes. The semicolon (;) is used to separate column families. The comma (,) is used to separate column qualifiers.
  - The column name and its data type must be included in '[]'.
  - Column names and their data types are separated by '->'.
  - If the data type of a specific column is not specified, the default data type (string) is used.
  - If **scan.caching** is not configured, the default value **1000** is used.
  - The user table must exist.
  - The index specified in the table must not exist.
  - If a column family named **d** exists in the user table, you must use the TableIndexer tool to build index data.

  After the preceding command is executed, the specified index is added to the table and is in INACTIVE state. This behavior is similar to the **addIndices()** API.

- **Creating index data for existing indexes in a user table**

  The command is as follows:

  hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer -Dtablename.to.index=tablename -Dindexnames.to.build ='idx_0#idx_1'

  The following parameters are required.

  - **tablename.to.index**: Indicates the name of a table for which an index is created.
  - **indexspecs.to.build**: Indicates an index name.
  - **scan.caching** (optional): Contains an integer value, indicating the number of cached rows to be transmitted to the scanner during data table scanning.

  The parameters in the preceding command are described as follows:

  - **idx_1**: Indicates an index name.

  NOTE

  - The pound key (#) is used to separate index names.
  - If **scan.caching** is not configured, the default value **1000** is used.
  - The user table must exist.

  After the preceding command is executed, the specified index is set to the ACTIVE state. Users can use them when scanning data.

- **Deleting the existing indexes and their data from a user table**

  The command is as follows:

  hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer -Dtablename.to.index=tablename -Dindexnames.to.drop='idx_0#idx_1'

  The following parameters are required.

  - **tablename.to.index**: Indicates the name of a table for which an index is created.
  - **indexnames.to.drop**: Indicates the name of the index that should be deleted with its data (must exist in the table).
  - **scan.caching** (optional): Contains an integer value, indicating the number of cached rows to be transmitted to the scanner during data table scanning.

  The parameters in the preceding command are described as follows:

  - **idx_1**: Indicates an index name.

  NOTE

  - The pound key (#) is used to separate index names.
  - If **scan.caching** is not configured, the default value **1000** is used.
  - The user table must exist.

  After the preceding command is executed, the specified index is deleted from the table.

- **Adding new indexes to user tables and building data based on existing data**

  The command is as follows:

  hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer -Dtablename.to.index=tablename -Dindexspecs.to.add='idx_0 => cf_0:[q_0-

> string],[q_1];cf_1:[q_2],[q_3]#idx_1 => cf_1:[q_4]' -Dindexnames.to.build='idx_0'

NOTE

- The parameters are the same as the previous ones.
- The user table must exist.
- The indexes specified in **indexspecs.to.add** must not exist in the table.
- The index names specified in **indexnames.to.build** must exist in the table or be part of the value of **indexspecs.to.add**.

After the preceding command is executed, all indexes specified in **indexspecs.to.add** will be added to this table, and index data will be built for all specified indexes using **indexnames.to.build**.

12.8.8.4 Migrating Index Data

Scenario

The indexes used in MRS 1.7 or later are incompatible with secondary indexes used by HBase in earlier MRS versions. Therefore, you need to perform the following operations to migrate index data from an earlier version (MRS 1.5 or earlier) to MRS 1.7 or later.

Prerequisites

1. During data migration, the cluster of the old version must be MRS 1.5 or earlier, and the cluster of the new version must be MRS 1.7 or later.
2. Before data migration, you must have old index data.
3. A cross-cluster mutual trust relationship must be configured and the inter-cluster replication function must be enabled for a security cluster. For a common cluster, only the inter-cluster replication function needs to be enabled

Procedure

Migrate user data from an old cluster to a new cluster. To migrate data, you need to manually synchronize data of the old and new clusters in a single table by export, distcp, and import.

For example, the current old cluster has a user table (**t1**, index name: **idx_t1**) and its corresponding index table (**t1_idx**). Perform the following operations to migrate data.

1. Export table data from the old cluster.
   ```
   hbase org.apache.hadoop.hbase.mapreduce.Export -Dhbase.mapreduce.include.deleted.rows=true <tableName> <path/for/data>
   ```
   - *<tableName>*: Indicates a table name, for example, **t1**.
   - *<path/for/data>*: Indicates the path for storing source data, for example, **/user/hbase/t1**.

   Example: **hbase org.apache.hadoop.hbase.mapreduce.Export -Dhbase.mapreduce.include.deleted.rows=true t1 /user/hbase/t1**
2. Copy the exported data to the new cluster as follows:

hadoop distcp *<path/for/data> hdfs://ActiveNameNodeIP:9820/*<path/for/newData>

- *<path/for/data>*: Indicates the path for storing source data in the old cluster, for example, **/user/hbase/t1**.
- *<path/for/newData>*: Indicates the path for storing source data in the new cluster, for example, **/user/hbase/t1**.

ActiveNameNodeIP indicates the IP address of the active NameNode in the new cluster.

Example: **hadoop distcp /user/hbase/t1 hdfs://192.168.40.2:9820/user/hbase/t1**

NOTE

- Manually copy the exported data to HDFS of the new cluster, for example, **/user/hbase/t1**.

3. Use the HBase table user of the new cluster to generate HFiles in the new cluster.

   hbase org.apache.hadoop.hbase.mapreduce.Import -Dimport.bulk.output=<path/for/hfiles> <tableName><path/for/newData>

   - *<path/for/hfiles>*: Indicates the path of the HFiles generated in the new cluster, for example, **/user/hbase/output_t1**.
   - *<tableName>*: Indicates a table name, for example, **t1**.
   - *<path/for/newData>*: Indicates the path for storing source data in the new cluster, for example, **/user/hbase/t1**.

   Example:

   hbase org.apache.hadoop.hbase.mapreduce.Import -Dimport.bulk.output=/user/hbase/output_t1 t1 /user/hbase/t1

4. Import the generated HFiles to the table in the new cluster.

   The command is as follows:

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles <path/for/hfiles> <tableName>

   - *<path/for/hfiles>*: Indicates the path of the HFiles generated in the new cluster, for example, **/user/hbase/output_t1**.
   - *<tableName>*: Indicates a table name, for example, **t1**.

   Example:

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles /user/hbase/output_t1 t1

   NOTE

   1. The preceding shows the process of migrating user data. You only need to perform the first three steps to migrate the index data of the old cluster and change the corresponding table name to an index table name (for example, **t1_idx**).
   2. Skip **4** when migrating index data.

5. Import index data to a table in the new cluster.

   a. Add an index the same as that of the user table of the previous version to the user table of the new cluster (the user table cannot contain a column family named **d**).

      The command is as follows:

      hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer -Dtablename.to.index=<tableName> -Dindexspecs.to.add=<indexspecs>

- *-Dtablename.to.index=<tableName>*: Indicates a table name, for example, **-Dtablename.to.index=t1**.
- **-Dindexspecs.to.add=<indexspecs>**: Indicates the mapping between an index name and a column, for example, **-Dindexspecs.to.add='idx_t1=>info:[name->String]'**.

Example:

hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer -Dtablename.to.index=t1 -Dindexspecs.to.add='idx_t1=>info:[name->String]'

NOTE

If a column family named **d** exists in the user table, you must use the TableIndexer tool to build index data.

b. Run the LoadIncrementalHFiles tool to load the index data of the old cluster to a table in the new cluster.

The command is as follows:

```
hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles </path/for/hfiles> <tableName>
```

- *</path/for/hfiles>*: Indicates the path of index data on HDFS. The path is the index generation path specified in **-Dimport.bulk.output**, for example, **/user/hbase/output_t1_idx**.
- *<tableName>*: Indicates a table name of the new cluster, for example, **t1**.

Example:

hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles /user/hbase/output_t1_idx t1

12.8.9 Configuring HBase DR

Scenario

HBase disaster recovery (DR), a key feature that is used to ensure high availability (HA) of the HBase cluster system, provides the real-time remote DR function for HBase. HBase DR provides basic O&M tools, including tools for maintaining and re-establishing DR relationships, verifying data, and querying data synchronization progress. To implement real-time DR, back up data of an HBase cluster to another HBase cluster. DR in the HBase table common data writing and BulkLoad batch data writing scenarios is supported.

NOTE

This section applies to MRS 3.*x* or later.

Prerequisites

- The active and standby clusters are successfully installed and started, and you have the administrator permissions on the clusters.
- Ensure that the network connection between the active and standby clusters is normal and ports are available.

- If the active cluster is deployed in security mode and is not managed by one FusionInsight Manager, cross-cluster trust relationship has been configured for the active and standby clusters.. If the active cluster is deployed in normal mode, no cross-cluster mutual trust is required.
- Cross-cluster replication has been configured for the active and standby clusters.
- Time is consistent between the active and standby clusters and the NTP service on the active and standby clusters uses the same time source.
- Mapping relationships between the names of all hosts in the active and standby clusters and IP addresses have been configured in the hosts files of all the nodes in the active and standby clusters and of the node where the active cluster client resides.
- The network bandwidth between the active and standby clusters is determined based on service volume, which cannot be less than the possible maximum service volume.
- The MRS versions of the active and standby clusters must be the same.
- The scale of the standby cluster must be greater than or equal to that of the active cluster.

Constraints

- Although DR provides the real-time data replication function, the data synchronization progress is affected by many factors, such as the service volume in the active cluster and the health status of the standby cluster. In normal cases, the standby cluster should not take over services. In extreme cases, system maintenance personnel and other decision makers determine whether the standby cluster takes over services according to the current data synchronization indicators.
- HBase clusters must be deployed in active/standby mode.
- Table-level operations on the DR table of the standby cluster are forbidden, such as modifying the table attributes and deleting the table. Misoperations on the standby cluster will cause data synchronization failure of the active cluster. As a result, table data in the standby cluster is lost.
- If the DR data synchronization function is enabled for HBase tables of the active cluster, the DR table structure of the standby cluster needs to be modified to ensure table structure consistency between the active and standby clusters during table structure modification.

Procedure

Configuring the common data writing DR parameters for the active cluster

Step 1 Log in to Manager of the active cluster.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations** and click **All Configurations**. The HBase configuration page is displayed.

Step 3 (Optional) **Table 12-168** describes the optional configuration items during HBase DR. You can set the parameters based on the description or use the default values.

Table 12-168 Optional configuration items

Navigation Path	Parameter	Default Value	Description
HMaster > Performance	hbase.master.logcleaner.ttl	600000	Specifies the retention period of HLog. If the value is set to **604800000** (unit: millisecond), the retention period of HLog is 7 days.
	hbase.master.cleaner.interval	60000	Interval for the HMaster to delete historical HLog files. The HLog that exceeds the configured period will be automatically deleted. You are advised to set it to the maximum value to save more HLogs.
RegionServer > Replication	replication.source.size.capacity	16777216	Maximum size of edits, in bytes. If the edit size exceeds the value, HLog edits will be sent to the standby cluster.
	replication.source.nb.capacity	25000	Maximum number of edits, which is another condition for triggering HLog edits to be sent to the standby cluster. After data in the active cluster is synchronized to the standby cluster, the active cluster reads and sends data in HLog according to this parameter value. This parameter is used together with **replication.source.size.capacity**.
	replication.source.maxretriesmultiplier	10	Maximum number of retries when an exception occurs during replication.
	replication.source.sleepforretries	1000	Retry interval (Unit: ms)
	hbase.regionserver.replication.handler.count	6	Number of replication RPC server instances on RegionServer

Configuring the BulkLoad batch data writing DR parameters for the active cluster

Step 4 Determine whether to enable the BulkLoad batch data writing DR function.

If yes, go to **Step 5**.

If no, go to **Step 8**.

Step 5 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations** and click **All Configurations**. The HBase configuration page is displayed.

Step 6 Search for **hbase.replication.bulkload.enabled** and change its value to **true** to enable the BulkLoad batch data writing DR function.

Step 7 Search for **hbase.replication.cluster.id** and change the HBase ID of the active cluster. The ID is used by the standby cluster to connect to the active cluster. The value can contain uppercase letters, lowercase letters, digits, and underscores (_), and cannot exceed 30 characters.

Restarting the HBase service and install the client

Step 8 Click **Save**. In the displayed dialog box, click **OK**. Restart the HBase service.

Step 9 In the active and standby clusters, choose **Cluster** > ***Name of the desired cluster*** > **Service** > **HBase** > **More** > **Download Client** to download the client and install it.

Adding the DR relationship between the active and standby clusters

Step 10 Log in as user **hbase** to the HBase shell page of the active cluster.

Step 11 Run the following command on HBase Shell to create the DR synchronization relationship between the active cluster HBase and the standby cluster HBase.

add_peer '*Standby cluster ID*', **CLUSTER_KEY => "***ZooKeeper service IP address in the standby cluster* **", CONFIG => {"hbase.regionserver.kerberos.principal" => "***Standby cluster RegionServer principal***", "hbase.master.kerberos.principal" => "***Standby cluster HMaster principal***"}**

- The standby cluster ID indicates the ID for the active cluster to recognize the standby cluster. Enter an ID. The value can be specified randomly. Digits are recommended.
- The ZooKeeper address of the standby cluster includes the service IP address of ZooKeeper, the port for listening to client connections, and the HBase root directory of the standby cluster on ZooKeeper.
- Search for **hbase.master.kerberos.principal** and **hbase.regionserver.kerberos.principal** in the HBase **hbase-site.xml** configuration file of the standby cluster.

For example, to add the DR relationship between the active and standby clusters, run the **add_peer** '*Standby cluster ID*', **CLUSTER_KEY => "*192.168.40.2,192.168.40.3,192.168.40.4:24002:/hbase*", CONFIG => {"hbase.regionserver.kerberos.principal" => "*hbase/hadoop.hadoop.com@HADOOP.COM*", "hbase.master.kerberos.principal" => "*hbase/hadoop.hadoop.com@HADOOP.COM*"}**

Step 12 (Optional) If the BulkLoad batch data write DR function is enabled, the HBase client configuration of the active cluster must be copied to the standby cluster.

- Create the **/hbase/replicationConf/*hbase.replication.cluster.id of the active cluster*** directory in the HDFS of the standby cluster.

- HBase client configuration file, which is copied to the **/hbase/replicationConf/*hbase.replication.cluster.id of the active cluster*** directory of the HDFS of the standby cluster.

  Example: **hdfs dfs -put HBase/hbase/conf/core-site.xml HBase/hbase/conf/hdfs-site.xml HBase/hbase/conf/yarn-site.xml hdfs://*NameNode IP:25000*/hbase/replicationConf/*source_cluster***

Enabling HBase DR to synchronize data

Step 13 Check whether a naming space exists in the HBase service instance of the standby cluster and the naming space has the same name as the naming space of the HBase table for which the DR function is to be enabled.

- If the same namespace exists, go to **Step 14**.
- If no, create a naming space with the same name in the HBase shell of the standby cluster and go to **Step 14**.

Step 14 In the HBase shell of the active cluster, run the following command as user **hbase** to enable the real-time DR function for the table data of the active cluster to ensure that the data modified in the active cluster can be synchronized to the standby cluster in real time.

You can only synchronize the data of one HTable at a time.

enable_table_replication '*table name***'**

NOTE

- If the standby cluster does not contain a table with the same name as the table for which real-time synchronization is to be enabled, the table is automatically created.
- If a table with the same name as the table for which real-time synchronization is to be enabled exists in the standby cluster, the structures of the two tables must be the same.
- If the encryption algorithm SMS4 or AES is configured for '*Table name*', the function for synchronizing data from the active cluster to the standby cluster cannot be enabled for the HBase table.
- If the standby cluster is offline or has tables with the same name but different structures, the DR function cannot be enabled.
- If the DR data synchronization function is enabled for some Phoenix tables in the active cluster, the standby cluster cannot have common HBase tables with the same names as the Phoenix tables in the active cluster. Otherwise, the DR function fails to be enabled or the tables with the names in the standby cluster cannot be used properly.
- If the DR data synchronization function is enabled for Phoenix tables in the active cluster, you need to enable the DR data synchronization function for the metadata tables of the Phoenix tables. The metadata tables include SYSTEM.CATALOG, SYSTEM.FUNCTION, SYSTEM.SEQUENCE, and SYSTEM.STATS.
- If the DR data synchronization function is enabled for HBase tables of the active cluster, after adding new indexes to HBase tables, you need to manually add secondary indexes to DR tables in the standby cluster to ensure secondary index consistency between the active and standby clusters.
- The HBase multi-instance function also supports DR. You need to modify the parameters on the HBase service instance that corresponds to the standby cluster and run the commands on the clients of multiple instances. When adding the DR relationship, you need to select the directory, such as **hbase1**, for ZooKeeper of the standby cluster to store HBase multi-instance data.

Step 15 (Optional) If HBase does not use Ranger, run the following command as user **hbase** in the HBase shell of the active cluster to enable the real-time permission to control data DR function for the HBase tables in the active cluster.

enable_table_replication 'hbase:acl'

Creating Users

Step 16 Log in to FusionInsight Manager of the standby cluster, choose **System** > **Permission** > **Role** > **Create Role** to create a role, and add the same permission for the standby data table to the role based on the permission of the HBase source data table of the active cluster.

Step 17 Choose **System** > **Permission** > **User** > **Create** to create a user. Set the **User Type** to **Human-Machine** or **Machine-Machine** based on service requirements and add the user to the created role. Access the HBase DR data of the standby cluster as the newly created user.

NOTE

- After the permission of the active HBase source data table is modified, to ensure that the standby cluster can properly read data, modify the role permission for the standby cluster.
- If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for HBase**.

Synchronizing the table data of the active cluster

Step 18 After HBase DR is configured and data synchronization is enabled, check whether tables and data exist in the active cluster and whether the historical data needs to be synchronized to the standby cluster.

- If yes, a table exists and data needs to be synchronized. Log in as the HBase table user to the node where the HBase client of the active cluster is installed and run the kinit username to authenticate the identity. The user must have the read and write permissions on tables and the execute permission on the **hbase:meta** table. Then go to **Step 19**.
- If no, no further action is required.

Step 19 The HBase DR configuration does not support automatic synchronization of historical data in tables. You need to back up the historical data of the active cluster and then manually restore the historical data in the standby cluster.

Manual recovery refers to the recovery of a single table, which can be performed through Export, DistCp, or Import.

To manually recover a single table, perform the following steps:

1. Export table data from the active cluster.

   hbase org.apache.hadoop.hbase.mapreduce.Export -Dhbase.mapreduce.include.deleted.rows=true *Table name Directory where the source data is stored*

   Example: **hbase org.apache.hadoop.hbase.mapreduce.Export -Dhbase.mapreduce.include.deleted.rows=true t1 /user/hbase/t1**

2. Copy the data that has been exported to the standby cluster.

   hadoop distcp *directory where the source data is stored on the active cluster* **hdfs://***ActiveNameNodeIP:8020/directory where the source data is stored on the standby cluster*

   ActiveNameNodeIP indicates the IP address of the active NameNode in the standby cluster.

Example: **hadoop distcp /user/hbase/t1 hdfs://192.168.40.2:8020/user/hbase/t1**

3. Import data to the standby cluster as the HBase table user of the standby cluster.

   On the HBase shell screen of the standby cluster, run the following command as user **hbase** to retain the data writing status:

   set_clusterState_active

   The command is run successfully if the following information is displayed:

   ```
   hbase(main):001:0> set_clusterState_active
   => true
   ```

   hbase org.apache.hadoop.hbase.mapreduce.Import -*Dimport.bulk.output=Directory where the output data is stored in the standby cluster Table name Directory where the source data is stored in the standby cluster*

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles *Directory where the output data is stored in the standby cluster Table name*

   Example:

   ```
   hbase(main):001:0> set_clusterState_active
   => true
   ```

   hbase org.apache.hadoop.hbase.mapreduce.Import -Dimport.bulk.output=/user/hbase/output_t1 t1 /user/hbase/t1

   hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles /user/hbase/output_t1 t1

Step 20 Run the following command on the HBase client to check the synchronized data of the active and standby clusters. After the DR data synchronization function is enabled, you can run this command to check whether the newly synchronized data is consistent.

hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication --starttime*=Start time* **--endtime***=End time Column family name ID of the standby cluster Table name*

NOTE

- The start time must be earlier than the end time.
- The values of **starttime** and **endtime** must be in the timestamp format. You need to run **date -d "2015-09-30 00:00:00" +%s to** change a common time format to a timestamp format.

Specify the data writing status for the active and standby clusters.

Step 21 On the HBase shell screen of the active cluster, run the following command as user **hbase** to retain the data writing status:

set_clusterState_active

The command is run successfully if the following information is displayed:

```
hbase(main):001:0> set_clusterState_active
=> true
```

Step 22 On the HBase shell screen of the standby cluster, run the following command as user **hbase** to retain the data read-only status:

set_clusterState_standby

The command is run successfully if the following information is displayed:

```
hbase(main):001:0> set_clusterState_standby
=> true
```

----End

Related Commands

Table 12-169 HBase DR

Operation	Command	Description
Set up a DR relationship.	add_peer'*Standby cluster ID*', CLUSTER_KEY => "*Standby cluster ZooKeeper service IP address*", CONFIG => {"hbase.regionserver.kerberos.principal" => "*Standby cluster RegionServer principal*", "hbase.master.kerberos.principal" => "*Standby cluster HMaster principal*"} **add_peer '1','zk1,zk2,zk3:2181:/hbase1'** **2181**: port number of ZooKeeper in the cluster	Set up the relationship between the active cluster and the standby cluster. If BulkLoad batch data write DR is enabled: • Create the **/hbase/replicationConf/*hbase.replication.cluster.id of the active cluster*** directory in the HDFS of the standby cluster. • HBase client configuration file, which is copied to the **/hbase/replicationConf/*hbase.replication.cluster.id of the active cluster*** directory of the HDFS of the standby cluster.
Remove the DR relationship.	**remove_peer** *'Standby cluster ID'* Example: **remove_peer '1'**	Remove standby cluster information from the active cluster.
Querying the DR Relationship	**list_peers**	Query standby cluster information (mainly Zookeeper information) in the active cluster.
Enable the real-time user table synchronization function.	**enable_table_replication** *'Table name'* Example: **enable_table_replication 't1'**	Synchronize user tables from the active cluster to the standby cluster.

Operation	Command	Description
Disable the real-time user table synchronization function.	**disable_table_replication** *'Table name'* Example: **disable_table_replication 't1'**	Do not synchronize user tables from the active cluster to the standby cluster.
Verify data of the active and standby clusters.	**bin/hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication --starttime=***Start time* **--endtime=***End time Column family name Standby cluster ID Table name*	Verify whether data of the specified table is the same between the active cluster and the standby cluster. The description of the parameters in this command is as follows: ● Start time: If start time is not specified, the default value **0** will be used. ● End time: If end time is not specified, the time when the current operation is submitted will be used by default. ● Table name: If a table name is not entered, all user tables for which the real-time synchronization function is enabled will be verified by default.
Switch the data writing status.	**set_clusterState_active** **set_clusterState_standby**	Specifies whether data can be written to the cluster HBase tables.

Operation	Command	Description
Add or update the active cluster HDFS configurations saved in the peer cluster.	**hdfs dfs -put -f HBase/hbase/conf/core-site.xml HBase/hbase/conf/hdfs-site.xml HBase/hbase/conf/yarn-site.xml hdfs://***Standby cluster* ***NameNode IP:PORT*****/hbase/replicationConf/***Active cluster***hbase.replication.cluster.id***	Enable DR for data including bulkload data. When HDFS parameters are modified in the active cluster, the modification cannot be automatically synchronized from the active cluster to the standby cluster. You need to manually run the command to synchronize configuration. The affected parameters are as follows: • fs.defaultFS • dfs.client.failover.proxy.provider.hacluster • dfs.client.failover.connection.retries.on.timeouts • dfs.client.failover.connection.retries For example, change **fs.defaultFS** to **hdfs://hacluster_sale**, HBase client configuration file, which is copied to the **/hbase/replicationConf/*hbase.replication.cluster.id of the active cluster*** directory of the HDFS of the standby cluster.

12.8.10 Configuring HBase Data Compression and Encoding

Scenario

HBase encodes data blocks in HFiles to reduce duplicate keys in KeyValues, reducing used space. Currently, the following data block encoding modes are supported: NONE, PREFIX, DIFF, FAST_DIFF, and ROW_INDEX_V1. NONE indicates that data blocks are not encoded. HBase also supports compression algorithms for HFile compression. The following algorithms are supported by default: NONE, GZ, SNAPPY, and ZSTD. NONE indicates that HFiles are not compressed.

The two methods are used on the HBase column family. They can be used together or separately.

Prerequisites

- You have installed an HBase client. For example, the client is installed in **opt/client**.

- If authentication has been enabled for HBase, you must have the corresponding operation permissions. For example, you must have the creation (C) or administration (A) permission on the corresponding namespace or higher-level items to create a table, and the creation (C) or administration (A) permission on the created table or higher-level items to modify a table. For details about how to grant permissions, see **Creating HBase Roles**.

Procedure

Setting data block encoding and compression algorithms during creation

- **Method 1: Using hbase shell**
  a. Log in to the node where the client is installed as the client installation user.
  b. Run the following command to go to the client directory:
     cd /opt/client
  c. Run the following command to configure environment variables:
     source bigdata_env
  d. If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:
     kinit *Component service user*
     For example, **kinit hbaseuser**.
  e. Run the following HBase client command:
     hbase shell
  f. Create a table.
     create '*t1***', {NAME => '***f1***', COMPRESSION => '***SNAPPY***', DATA_BLOCK_ENCODING => '***FAST_DIFF***'}**

     NOTE

     - *t1*: indicates the table name.
     - *f1*: indicates the column family name.
     - *SNAPPY*: indicates the column family uses the SNAPPY compression algorithm.
     - *FAST_DIFF*: indicates FAST_DIFF is used for encoding.
     - The parameter in the braces specifies the column family. You can specify multiple column families using multiple braces and separate them by commas (,). For details about table creation statements, run the **help 'create'** statement in the HBase shell.
- **Method 2: Using Java APIs**
  The following code snippet shows only how to set the encoding and compression modes of a column family when creating a table. For complete code for creating a table and how to use the code to create a table, see "HBase Development Guide" > "Modifying a Table" in .

```
TableDescriptorBuilder htd = TableDescriptorBuilder.newBuilder(TableName.valueOf("t1"));// Create a descriptor for table t1.
ColumnFamilyDescriptorBuilder hcd = ColumnFamilyDescriptorBuilder.newBuilder(Bytes.toBytes("f1"));// Create a builder for column family f1.
```

```
hcd.setDataBlockEncoding(DataBlockEncoding.FAST_DIFF);// Set the encoding mode of column family f1 to FAST_DIFF.
hcd.setCompressionType(Compression.Algorithm.SNAPPY);// Set the compression algorithm of column family f1 to SNAPPY.
htd.setColumnFamily(hcd.build())// Add the column family f1 to the descriptor of table t1.
```

Setting or modifying the data block encoding mode and compression algorithm for an existing table

- **Method 1: Using hbase shell**
  a. Log in to the node where the client is installed as the client installation user.
  b. Run the following command to go to the client directory:
     cd /opt/client
  c. Run the following command to configure environment variables:
     source bigdata_env
  d. If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:
     kinit *Component service user*
     For example, **kinit hbaseuser**.
  e. Run the following HBase client command:
     hbase shell
  f. Run the following command to modify the table:
     alter '*t1***', {NAME => '***f1***', COMPRESSION => '***SNAPPY***', DATA_BLOCK_ENCODING => '***FAST_DIFF***'}**
- **Method 2: Using Java APIs**
  The following code snippet shows only how to modify the encoding and compression modes of a column family in an existing table. For complete code for modifying a table and how to use the code to modify a table, see "HBase Development Guide".

```
TableDescriptor htd = admin.getDescriptor(TableName.valueOf("t1"));// Obtain the descriptor of table t1.
ColumnFamilyDescriptor originCF = htd.getColumnFamily(Bytes.toBytes("f1"));// Obtain the descriptor of column family f1.
builder.ColumnFamilyDescriptorBuilder hcd = ColumnFamilyDescriptorBuilder.newBuilder(originCF);// Create a builder based on the existing column family attributes.
hcd.setDataBlockEncoding(DataBlockEncoding.FAST_DIFF);// Change the encoding mode of the column family to FAST_DIFF.
hcd.setCompressionType(Compression.Algorithm.SNAPPY);// Change the compression algorithm of the column family to SNAPPY.
admin.modifyColumnFamily(TableName.valueOf("t1"), hcd.build());// Submit to the server to modify the attributes of column family f1.
```

  After the modification, the encoding and compression modes of the existing HFile will take effect after the next compaction.

12.8.11 Performing an HBase DR Service Switchover

Scenario

The system administrator can configure HBase cluster DR to improve system availability. If the active cluster in the DR environment is faulty and the connection

to the HBase upper-layer application is affected, you need to configure the standby cluster information for the HBase upper-layer application so that the application can run in the standby cluster.

NOTE

This section applies to MRS 3.*x* or later.

Impact on the System

After a service switchover, data written to the standby cluster is not synchronized to the active cluster by default. Add the active cluster is recovered, the data newly generated in the standby cluster needs to be synchronized to the active cluster by backup and recovery. If automatic data synchronization is required, you need to switch over the active and standby HBase DR clusters.

Procedure

Step 1 Log in to FusionInsight Manager of the standby cluster.

Step 2 Download and install the HBase client.

Step 3 On the HBase client of the standby cluster, run the following command as user **hbase** to enable the data writing status in the standby cluster.

kinit hbase

hbase shell

set_clusterState_active

The command is run successfully if the following information is displayed:

```
hbase(main):001:0> set_clusterState_active
=> true
```

Step 4 Check whether the original configuration files **hbase-site.xml**, **core-site.xml**, and **hdfs-site.xml** of the HBase upper-layer application are modified to adapt to the application running.

- If yes, update the related content to the new configuration file and replace the old configuration file.
- If no, use the new configuration file to replace the original configuration file of the HBase upper-layer application.

Step 5 Configure the network connection between the host where the HBase upper-layer application is located and the standby cluster.

NOTE

If the host where the client is installed is not a node in the cluster, configure network connections for the client to prevent errors when you run commands on the client.

1. Ensure that the host where the client is installed can communicate with the hosts listed in the **hosts** file in the directory where the client installation package is decompressed.
2. If the host where the client is located is not a node in the cluster, you need to set the mapping between the host name and the IP address (service plan) in

the /etc/hosts file on the host. The host names and IP addresses must be mapped one by one.

Step 6 Set the time of the host where the HBase upper-layer application is located to be the same as that of the standby cluster. The time difference must be less than 5 minutes.

Step 7 Check the authentication mode of the active cluster.

- If the security mode is used, go to **Step 8**.
- If the normal mode is used, no further action is required.

Step 8 Obtain the **keytab** and **krb5.conf** configuration files of the HBase upper-layer application user.

1. On FusionInsight Manager of the standby cluster, choose **System** > **Permission** > **User**.
2. Locate the row that contains the target user, click **More** > **Download Authentication Credential** in the **Operation** column, and download the **keytab** file to the local PC.
3. Decompress the package to obtain **user.keytab** and **krb5.conf**.

Step 9 Use the **user.keytab** and **krb5.conf** files to replace the original files in the HBase upper-layer application.

Step 10 Stop upper-layer applications.

Step 11 Determine whether to switch over the active and standby HBase clusters. If the switchover is not performed, data will not be synchronized.

- If yes, switch over the active and standby HBase DR clusters. For details, see **Performing an HBase DR Active/Standby Cluster Switchover**. Then, go to **Step 12**.
- If no, go to **Step 12**.

Step 12 Start the upper-layer services.

----End

12.8.12 Performing an HBase DR Active/Standby Cluster Switchover

Scenario

The HBase cluster in the current environment is a DR cluster. Due to some reasons, the active and standby clusters need to be switched over. That is, the standby cluster becomes the active cluster, and the active cluster becomes the standby cluster.

NOTE

This section applies to MRS 3.*x* or later.

Impact on the System

After the active and standby clusters are switched over, data cannot be written to the original active cluster, and the original standby cluster becomes the active cluster to take over upper-layer services.

Procedure

Ensuring that upper-layer services are stopped

Step 1 Ensure that the upper-layer services have been stopped. If not, perform operations by referring to **Performing an HBase DR Service Switchover**.

Disabling the write function of the active cluster

Step 2 Download and install the HBase client.

Step 3 On the HBase client of the standby cluster, run the following command as user **hbase** to disable the data write function of the standby cluster:

kinit hbase

hbase shell

set_clusterState_standby

The command is run successfully if the following information is displayed:

```
hbase(main):001:0> set_clusterState_standby
=> true
```

Checking whether the active/standby synchronization is complete

Step 4 Run the following command to ensure that the current data has been synchronized (SizeOfLogQueue=0 and SizeOfLogToReplicate=0 are required). If the values are not 0, wait and run the following command repeatedly until the values are 0.

status 'replication'

Disabling synchronization between the active and standby clusters

Step 5 Query all synchronization clusters and obtain the value of **PEER_ID**.

list_peers

Step 6 Delete all synchronization clusters.

remove_peer *'Standby cluster ID'*

Example:

remove_peer *'1'*

Step 7 Query all synchronized tables.

list_replicated_tables

Step 8 Disable all synchronized tables queried in the preceding step.

disable_table_replication *'Table name'*

Example:

disable_table_replication *'t1'*

Performing an active/standby switchover

Step 9 Reconfigure HBase DR. For details, see **Configuring HBase DR**.

----End

12.8.13 Community BulkLoad Tool

The Apache HBase official website provides the function of importing data in batches. For details, see the description of the **Import** and **ImportTsv** tools at **http://hbase.apache.org/2.2/book.html#tools**.

12.8.14 Configuring the MOB

Scenario

In the actual application scenario, data in various sizes needs to be stored, for example, image data and documents. Data whose size is smaller than 10 MB can be stored in HBase. HBase can yield the best read-and-write performance for data whose size is smaller than 100 KB. If the size of data stored in HBase is greater than 100 KB or even reaches 10 MB and the same number of data files are inserted, the total data amount is large, causing frequent compaction and split, high CPU consumption, high disk I/O frequency, and low performance.

MOB data (100 KB to 10 MB data) is stored in a file system (such as the HDFS) in the HFile format. Files are centrally managed using the expiredMobFileCleaner and Sweeper tools. The addresses and size of files are stored in the HBase store as values. This greatly decreases the compaction and split frequency in HBase and improves performance.

The MOB function of HBase is enabled by default. For details about related configuration items, see **Table 12-170**. To use the MOB function, you need to specify the MOB mode for storing data in the specified column family when creating a table or modifying table attributes.

NOTE

This section applies to MRS 3.*x* or later.

Configuration Description

To enable the HBase MOB function, you need to specify the MOB mode for storing data in the specified column family when creating a table or modifying table attributes.

Use code to declare that the MOB mode for storing data is used:

```
HColumnDescriptor hcd = new HColumnDescriptor("f");
hcd.setMobEnabled(true);
```

Use code to declare that the MOB mode for storing data is used, the unit of MOB_THRESHOLD is byte:

```
hbase(main):009:0> create 't3',{NAME => 'd', MOB_THRESHOLD => '102400', IS_MOB => 'true'}

0 row(s) in 0.3450 seconds

=> Hbase::Table - t3
hbase(main):010:0> describe 't3'
Table t3 is ENABLED


t3


COLUMN FAMILIES DESCRIPTION


{NAME => 'd', MOB_THRESHOLD => '102400', VERSIONS => '1', KEEP_DELETED_CELLS => 'FALSE',
DATA_BLOCK_ENCODING => 'NONE',
TTL => 'FOREVER', MIN_VERSIONS => '0', REPLICATION_SCOPE => '0', BLOOMFILTER => 'ROW',
IN_MEMORY => 'false', IS_MOB => 'true', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE =>
'65536'}

1 row(s) in 0.0170 seconds
```

Navigation path for setting parameters:

On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations** > **All Configurations**. Enter a parameter name in the search box.

Table 12-170 Parameter description

Parameter	Description	Default Value
hbase.mob.file.cache.size	Size of the opened file handle cache. If this parameter is set to a large value, more file handles can be cached, reducing the frequency of opening and closing files. However, if this parameter is set to a large value, too many file handles will be opened. The default value is **1000**. This parameter is configured on the ResionServer.	1000
hbase.mob.cache.evict.period	Expiration time of cached MOB files in the MOB cache, in seconds.	3600
hbase.mob.cache.evict.remain.ratio	Ratio of the number of retained files after MOB cache reclamation to the number of cached files. **hbase.mob.cache.evict.remain.ratio** is an algorithm factor. When the number of cached MOB files reaches the product of **hbase.mob.file.cache.size** **hbase.mob.cache.evict.remain.ratio**, cache reclamation is triggered.	0.5

Parameter	Description	Default Value
hbase.master.mob.ttl.cleaner.period	Interval for deleting expired files, in seconds. The default value is one day (86,400 seconds). **NOTE** If the validity period of an MOB file expires, that is, the file has been created for more than 24 hours, the MOB file will be deleted by the tool for deleting expired MOB files.	86400

12.8.15 Configuring Secure HBase Replication

Scenario

This topic provides the procedure to configure the secure HBase replication during cross-realm Kerberos setup in security mode.

Prerequisites

- Mapping for all the FQDNs to their realms should be defined in the Kerberos configuration file.
- The passwords and keytab files of **ONE.COM** and **TWO.COM** must be the same.

Procedure

Step 1 Create krbtgt principals for the two realms.

For example, if you have two realms called **ONE.COM** and **TWO.COM**, you need to add the following principals: **krbtgt/ONE.COM@TWO.COM** and **krbtgt/TWO.COM@ONE.COM**.

Add these two principals at both realms.

```
kadmin: addprinc -e "<enc_type_list>" krbtgt/ONE.COM@TWO.COM
kadmin: addprinc -e "<enc_type_list>" krbtgt/TWO.COM@ONE.COM
```

NOTE

There must be at least one common keytab mode between these two realms.

Step 2 Add rules for creating short names in Zookeeper.

Dzookeeper.security.auth_to_local is a parameter of the ZooKeeper server process. Following is an example rule that illustrates how to add support for the realm called **ONE.COM**. The principal has two members (such as **service/instance@ONE.COM**).

```
Dzookeeper.security.auth_to_local=RULE:[2:\$1@\$0](.*@\\QONE.COM\\E$)s/@\\QONE.COM\\E$//DEFAULT
```

The above code example adds support for the **ONE.COM** realm in a different realm. Therefore, in the case of replication, you must add a rule for the master cluster realm in the slave cluster realm. **DEFAULT** is for defining the default rule.

Step 3 Add rules for creating short names in the Hadoop processes.

The following is the **hadoop.security.auth_to_local** property in the **core-site.xml** file in the slave cluster HBase processes. For example, to add support for the **ONE.COM** realm:

```
<property>
<name>hadoop.security.auth_to_local</name>
<value>RULE:[2:$1@$0](.*@\QONE.COM\E$)s/@\QONE.COM\E$//DEFAULT</value>
</property>
```

NOTE

If replication for bulkload data is enabled, then the same property for supporting the slave realm needs to be added in the **core-site.xml** file in the master cluster HBase processes.

Example:

```
<property>
<name>hadoop.security.auth_to_local</name>
<value>RULE:[2:$1@$0](.*@\QTWO.COM\E$)s/@\QTWO.COM\E$//DEFAULT</value>
</property>
```

----End

12.8.16 Configuring Region In Transition Recovery Chore Service

Scenario

In a faulty environment, there are possibilities that a region may be stuck in transition for longer duration due to various reasons like slow region server response, unstable network, ZooKeeper node version mismatch. During region transition, client operation may not work properly as some regions will not be available.

Configuration

A chore service should be scheduled at HMaster to identify and recover regions that stay in the transition state for a long time.

The following table describes the parameters for enabling this function.

Table 12-171 Parameters

Parameter	Description	Default Value
hbase.region.assignment.auto.recovery.enabled	Configuration parameter used to enable/disable the region assignment recovery thread feature.	true

12.8.17 Using a Secondary Index

Scenario

HIndex enables HBase indexing based on specific column values, making the retrieval of data highly efficient and fast.

Constraints

- Column families are separated by semicolons (;).
- Columns and data types must be contained in square brackets ([]).
- The column data type is specified by using -> after the column name.
- If the column data type is not specified, the default data type (string) is used.
- The number sign (#) is used to separate two index details.
- The following is an optional parameter:
  -Dscan.caching: number of cached rows when the data table is scanned.
  The default value is set to 1000.
- Indexes are created for a single region to repair damaged indexes.
  This function is not used to generate new indexes.

Procedure

Step 1 Install the HBase client. For details, see **Using an HBase Client**.

Step 2 Go to the client installation directory, for example, **/opt/client**.

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the following command to access HIndex:

hbase org.apache.hadoop.hbase.hindex.mapreduce.TableIndexer

Table 12-172 Common HIndex commands

Description	Command
Add Index	TableIndexer-Dtablename.to.index=table1-Dindexspecs.to.add='IDX1=>cf1:[q1->datatype],[q2],[q3];cf2:[q1->datatype],[q2->datatype]#IDX2=>cf1:[q5]'
Create Index	TableIndexer -Dtablename.to.index=table1 -Dindexnames.to.build='IDX1#IDX2'

Description	Command
Delete Index	TableIndexer -Dtablename.to.index=table1 -Dindexnames.to.drop='IDX1#IDX2'
Disable Index	TableIndexer -Dtablename.to.index=table1 -Dindexnames.to.disable='IDX1#IDX2'
Add and Create Index	TableIndexer -Dtablename.to.index=table1 -Dindexspecs.to.add='IDX1=>cf1:[q1->datatype],[q2],[q3];cf2:[q1->datatype],[q2->datatype]#IDX2=>cf1:[q5]' -Dindexnames.to.build='IDX1'
Create Index for a Single Region	TableIndexer -Dtablename.to.index=table1 -Dregion.to.index=regionEncodedName -Dindexnames.to.build='IDX1#IDX2'

NOTE

- **IDX1**: indicates the index name.
- **cf1**: indicates the column family name.
- **q1**: indicates the column name.
- **datatype**: indicates the data type, including String, Integer, Double, Float, Long, Short, Byte and Char.

----End

12.8.18 HBase Log Overview

Log Description

Log path: The default storage path of HBase logs is **/var/log/Bigdata/hbase/***Role name*.

- HMaster: **/var/log/Bigdata/hbase/hm** (run logs) and **/var/log/Bigdata/audit/hbase/hm** (audit logs)
- RegionServer: **/var/log/Bigdata/hbase/rs** (run logs) and **/var/log/Bigdata/audit/hbase/rs** (audit logs)
- ThriftServer: **/var/log/Bigdata/hbase/ts2** (run logs, **ts2** is the instance name) and **/var/log/Bigdata/audit/hbase/ts2** (audit logs, **ts2** is the instance name)

Log archive rule: The automatic log compression and archiving function of HBase is enabled. By default, when the size of a log file exceeds 30 MB, the log file is automatically compressed. The naming rule of a compressed log file is as follows: <*Original log name*>-<*yyyy-mm-dd_hh-mm-ss*>.[*ID*].**log.zip** A maximum of 20 latest compressed files are reserved. The number of compressed files can be configured on the Manager portal.

Table 12-173 HBase log list

Type	Name	Description
Run logs	hbase-<SSH_USER>-<process_name>-<hostname>.log	HBase system log that records the startup time, startup parameters, and most logs generated when the HBase system is running.
	hbase-<SSH_USER>-<process_name>-<hostname>.out	Log that records the HBase running environment information.
	<process_name>-<SSH_USER>-<DATE>-<PID>-gc.log	Log that records HBase junk collections.
	checkServiceDetail.log	Log that records whether the HBase service starts successfully.
	hbase.log	Log generated when the HBase service health check script and some alarm check scripts are executed.
	sendAlarm.log	Log that records alarms reported after execution of HBase alarm check scripts.
	hbase-haCheck.log	Log that records the active and standby status of HMaster
	stop.log	Log that records the startup and stop processes of HBase.
Audit logs	hbase-audit-<process_name>.log	Log that records HBase security audit.

Log Level

Table 12-174 describes the log levels supported by HBase. The priorities of log levels are FATAL, ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-174 Log levels

Level	Description
FATAL	Logs of this level record fatal error information about the current event processing that may result in a system crash.

Level	Description
ERROR	Logs of this level record error information about the current event processing, which indicates that system running is abnormal.
WARN	Logs of this level record abnormal information about the current event processing. These abnormalities will not result in system faults.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of the HBase service. For details, see **Modifying Cluster Service Configuration Parameters**.

Step 2 On the left menu bar, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

NOTE

The configurations take effect immediately without the need to restart the service.

----End

Log Formats

The following table lists the HBase log formats.

Table 12-175 Log formats

Type	Component	Format	Example
Run logs	HMaster	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2020-01-19 16:04:53,558 | INFO | main | env:HBASE_THRIFT_OPTS= | org.apache.hadoop.hbase.util.ServerCommandLine.logProcessInfo(ServerCommandLine.java:113)

Type	Component	Format	Example
	RegionServer	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2020-01-19 16:05:18,589 | INFO | regionserver16020-SendThread(linux-k6da:2181) | Client will use GSSAPI as SASL mechanism. | org.apache.zookeeper.client.ZooKeeperSaslClient$1.run(ZooKeeperSaslClient.java:285)
	ThriftServer	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2020-02-16 09:42:55,371 | INFO | main | loaded properties from hadoop-metrics2.properties | org.apache.hadoop.metrics2.impl.MetricsConfig.loadFirst(MetricsConfig.java:111)
Audit logs	HMaster	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2020-02-16 09:42:40,934 | INFO | master:linux-k6da:16000 | Master: [master:linux-k6da:16000] start operation called. | org.apache.hadoop.hbase.master.HMaster.run(HMaster.java:581)
	RegionServer	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2020-02-16 09:42:51,063 | INFO | main | RegionServer: [regionserver16020] start operation called. | org.apache.hadoop.hbase.regionserver.HRegionServer.startRegionServer(HRegionServer.java:2396)
	ThriftServer	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2020-02-16 09:42:55,512 | INFO | main | thrift2 server start operation called. | org.apache.hadoop.hbase.thrift2.ThriftServer.main(ThriftServer.java:421)

12.8.19 HBase Performance Tuning

12.8.19.1 Improving the BulkLoad Efficiency

Scenario

BulkLoad uses MapReduce jobs to directly generate files that comply with the internal data format of HBase, and then loads the generated StoreFiles to a running cluster. Compared with HBase APIs, BulkLoad saves more CPU and network resources.

ImportTSV is an HBase table data loading tool.

NOTE

This section applies to MRS 3.*x* and later versions.

Prerequisites

When using BulkLoad, the output path of the file has been specified using the **Dimporttsv.bulk.output** parameter.

Procedure

Add the following parameter to the BulkLoad command when performing a batch loading task:

Table 12-176 Parameter for improving BulkLoad efficiency

Parameter	Description	Value
-Dimporttsv.mapper.class	The construction of key-value pairs is moved from the user-defined mapper to reducer to improve performance. The mapper only needs to send the original text in each row to the reducer. The reducer parses the record in each row and creates a key-value) pair. **NOTE** When this parameter is set to **org.apache.hadoop.hbase.mapreduce.TsvImporterByteMapper**, this parameter is used only when the batch loading command without the *HBASE_CELL_VISIBILITY OR HBASE_CELL_TTL* option is executed. The **org.apache.hadoop.hbase.mapreduce.TsvImporterByteMapper** provides better performance.	org.apache.hadoop.hbase.mapreduce.TsvImporterByteMapper and org.apache.hadoop.hbase.mapreduce.TsvImporterTextMapper

12.8.19.2 Improving Put Performance

Scenario

In the scenario where a large number of requests are continuously put, setting the following two parameters to **false** can greatly improve the Put performance.

- **hbase.regionserver.wal.durable.sync**
- **hbase.regionserver.hfile.durable.sync**

When the performance is improved, there is a low probability that data is lost if three DataNodes are faulty at the same time. Exercise caution when configuring the parameters in scenarios that have high requirements on data reliability.

NOTE

This section applies to MRS 3.*x* and later versions.

Procedure

Navigation path for setting parameters:

On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Configurations** > **All Configurations**. Enter the parameter name in the search box, and change the value.

Table 12-177 Parameters for improving put performance

Parameter	Description	Value
hbase.wal.hsync	Specifies whether to enable WAL file durability to make the WAL data persistence on disks. If this parameter is set to **true**, the performance is affected because each WAL file is synchronized to the disk by the Hadoop fsync.	false
hbase.hfile.hsync	Specifies whether to enable the HFile durability to make data persistence on disks. If this parameter is set to true, the performance is affected because each Hfile file is synchronized to the disk by the Hadoop fsync.	false

12.8.19.3 Optimizing Put and Scan Performance

Scenario

HBase has many configuration parameters related to read and write performance. The configuration parameters need to be adjusted based on the read/write request loads. This section describes how to optimize read and write performance by modifying the RegionServer configurations.

NOTE

This section applies to MRS 3.*x* and later versions.

Procedure

- JVM GC parameters

  Suggestions on setting the RegionServer **GC_OPTS** parameter:

  - Set **-Xms** and **-Xmx** to the same value based on your needs. Increasing the memory can improve the read and write performance. For details, see the description of **hfile.block.cache.size** in **Table 12-179** and **hbase.regionserver.global.memstore.size** in **Table 12-178**.
  - Set **-XX:NewSize** and **-XX:MaxNewSize** to the same value. You are advised to set the value to **512M** in low-load scenarios and **2048M** in high-load scenarios.
  - Set **X-XX:CMSInitiatingOccupancyFraction** to be less than and equal to 90, and it is calculated as follows: **100 x (hfile.block.cache.size + hbase.regionserver.global.memstore.size + 0.05)**.
  - **-XX:MaxDirectMemorySize** indicates the non-heap memory used by the JVM. You are advised to set this parameter to **512M** in low-load scenarios and **2048M** in high-load scenarios.

    NOTE

    The **-XX:MaxDirectMemorySize** parameter is not used by default. If you need to set this parameter, add it to the **GC_OPTS** parameter.

- Put parameters

  RegionServer processes the data of the put request and writes the data to memstore and HLog.

  - When the size of memstore reaches the value of **hbase.hregion.memstore.flush.size**, memstore is updated to HDFS to generate HFiles.
  - Compaction is triggered when the number of HFiles in the column cluster of the current region reaches the value of **hbase.hstore.compaction.min**.
  - If the number of HFiles in the column cluster of the current region reaches the value of **hbase.hstore.blockingStoreFiles**, the operation of refreshing the memstore and generating HFiles is blocked. As a result, the put request is blocked.

Table 12-178 Put parameters

Parameter	Description	Default Value
hbase.wal.hsync	Indicates whether each WAL is persistent to disks. For details, see **Improving Put Performance**.	true
hbase.hfile.hsync	Indicates whether HFile write operations are persistent to disks. For details, see **Improving Put Performance**.	true

Parameter	Description	Default Value
hbase.hregion.memstore.flush.size	If the size of MemStore (unit: Byte) exceeds a specified value, MemStore is flushed to the corresponding disk. The value of this parameter is checked by each thread running **hbase.server.thread.wakefrequency**. It is recommended that you set this parameter to an integer multiple of the HDFS block size. You can increase the value if the memory is sufficient and the put load is heavy.	134217728
hbase.regionserver.global.memstore.size	Updates the size of all MemStores supported by the RegionServer before locking or forcible flush. It is recommended that you set this parameter to **hbase.hregion.memstore.flush.size x Number of regions with active writes/ RegionServer GC -Xmx**. The default value is **0.4**, indicating that 40% of RegionServer GC -Xmx is used.	0.4
hbase.hstore.flusher.count	Indicates the number of memstore flush threads. You can increase the parameter value in heavy-put-load scenarios.	2
hbase.regionserver.thread.compaction.small	Indicates the number of small compaction threads. You can increase the parameter value in heavy-put-load scenarios.	10

Parameter	Description	Default Value
hbase.hstore.blockingStoreFiles	If the number of HStoreFile files in a Store exceeds the specified value, the update of the HRegion will be locked until a compression is completed or the value of **base.hstore.blockingWaitTime** is exceeded. Each time MemStore is flushed, a StoreFile file is written into MemStore. Set this parameter to a larger value in heavy-put-load scenarios.	15

- Scan parameters

  Table 12-179 Scan parameters

Parameter	Description	Default Value
hbase.client.scanner.timeout.period	Client and RegionServer parameters, indicating the lease timeout period of the client executing the scan operation. You are advised to set this parameter to an integer multiple of 60000 ms. You can set this parameter to a larger value when the read load is heavy. The unit is milliseconds.	60000
hfile.block.cache.size	Indicates the data cache percentage in the RegionServer GC –Xmx. You can increase the parameter value in heavy-read-load scenarios, in order to improve cache hit ratio and performance. It indicates the percentage of the maximum heap (-Xmx setting) allocated to the block cache of HFiles or StoreFiles.	When offheap is disabled, the default value is **0.25**. When offheap is enabled, the default value is **0.1**.

- Handler parameters

Table 12-180 Handler parameters

Parameter	Description	Default Value
hbase.regionserver.handler.count	Indicates the number of RPC server instances on RegionServer. The recommended value ranges from 200 to 400.	200
hbase.regionserver.metahandler.count	Indicates the number of program instances for processing prioritized requests. The recommended value ranges from 200 to 400.	200

12.8.19.4 Improving Real-time Data Write Performance

Scenario

Scenarios where data needs to be written to HBase in real time, or large-scale and consecutive put scenarios

NOTE

This section applies to MRS 3.*x* and later versions.

Prerequisites

The HBase put or delete interface can be used to save data to HBase.

Procedure

- **Data writing server tuning**

  Parameter portal:

  Go to the **All Configurations** page of the HBase service. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-181 Configuration items that affect real-time data writing

Parameter	Description	Default Value
hbase.wal.hsync	Controls the synchronization degree when HLogs are written to the HDFS. If the value is **true**, HDFS returns only when data is written to the disk. If the value is **false**, HDFS returns when data is written to the OS cache. Set the parameter to **false** to improve write performance.	true
hbase.hfile.hsync	Controls the synchronization degree when HFiles are written to the HDFS. If the value is **true**, HDFS returns only when data is written to the disk. If the value is **false**, HDFS returns when data is written to the OS cache. Set the parameter to **false** to improve write performance.	true

Parameter	Description	Default Value
GC_OPTS	You can increase HBase memory to improve HBase performance because read and write operations are performed in HBase memory. **HeapSize** and **NewSize** need to be adjusted. When you adjust **HeapSize**, set **Xms** and **Xmx** to the same value to avoid performance problems when JVM dynamically adjusts **HeapSize**. Set **NewSize** to 1/8 of **HeapSize**. • **HMaster**: If HBase clusters enlarge and the number of Regions grows, properly increase the **GC_OPTS** parameter value of the HMaster. • **RegionServer**: A RegionServer needs more memory than an HMaster. If sufficient memory is available, increase the **HeapSize** value. **NOTE** When the value of **HeapSize** for the active HMaster is 4 GB, the HBase cluster can support 100,000 regions. Empirically, each time 35,000 regions are added to the cluster, the value of **HeapSize** must be increased by 2 GB. It is recommended that the value of **HeapSize** for the active HMaster not exceed 32 GB.	• HMaster -server -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=512M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=512M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M

Parameter	Description	Default Value
		• Region Server -server -Xms6G -Xmx6G -XX:NewSize=1024M -XX:MaxNewSize=1024M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=512M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M

Parameter	Description	Default Value
hbase.regionserver.handler.count	Indicates the number of RPC server instances started on RegionServer. If the parameter is set to an excessively large value, threads will compete fiercely. If the parameter is set to an excessively small value, requests will be waiting for a long time in RegionServer, reducing the processing capability. You can add threads based on resources. It is recommended that the value be set to **100** to **300** based on the CPU usage.	200
hbase.hregion.max.filesize	Indicates the maximum size of an HStoreFile, in bytes. If the size of any HStoreFile exceeds the value of this parameter, the managed Hregion is divided into two parts.	10737418240
hbase.hregion.memstore.flush.size	On the RegionServer, when the size of memstore that exists in memory of write operations exceeds **memstore.flush.size**, MemStoreFlusher performs the Flush operation to write the memstore to the corresponding store in the format of HFile. If RegionServer memory is sufficient and active Regions are few, increase the parameter value and reduce compaction times to improve system performance. The Flush operation may be delayed after it takes place. Write operations continue and memstore keeps increasing during the delay. The maximum size of memstore is: **memstore.flush.size** x **hbase.hregion.memstore.block.multiplier**. When the memstore size exceeds the maximum value, write operations are blocked. Properly increasing the value of **hbase.hregion.memstore.block.multiplier** can reduce the blocks and make performance become more stable. Unit: byte	134217728

Parameter	Description	Default Value
hbase.regionserver.global.memstore.size	Updates the size of all MemStores supported by the RegionServer before locking or forcible flush. On the RegionServer, the MemStoreFlusher thread performs the flush. The thread regularly checks memory occupied by write operations. When the total memory volume occupied by write operations exceeds the threshold, MemStoreFlusher performs the flush. Larger memstore will be flushed first and then smaller ones until the occupied memory is less than the threshold. Threshold = hbase.regionserver.global.memstore.size x hbase.regionserver.global.memstore.size.lower.limit x HBase_HEAPSIZE **NOTE** The sum of the parameter value and the value of **hfile.block.cache.size** cannot exceed 0.8, that is, memory occupied by read and write operations cannot exceed 80% of **HeapSize**, ensuring stable running of other operations.	0.4

Parameter	Description	Default Value
hbase.hstore.blockingStoreFiles	Check whether the number of files is larger than the value of **hbase.hstore.blockingStoreFiles** before you flush regions. If it is larger than the value of **hbase.hstore.blockingStoreFiles**, perform a compaction and configure **hbase.hstore.blockingWaitTime** to 90s to make the flush delay for 90s. During the delay, write operations continue and the memstore size keeps increasing and exceeds the threshold (**memstore.flush.size** x **hbase.hregion.memstore.block.multiplier**), blocking write operations. After compaction is complete, a large number of writes may be generated. As a result, the performance fluctuates sharply. Increase the value of **hbase.hstore.blockingStoreFiles** to reduce block possibilities.	15
hbase.regionserver.thread.compaction.throttle	The compression whose size is greater than the value of this parameter is executed by the large thread pool. The unit is bytes. Indicates a threshold of a total file size for compaction during a Minor Compaction. The total file size affects execution duration of a compaction. If the total file size is large, other compactions or flushes may be blocked.	1610612736
hbase.hstore.compaction.min	Indicates the minimum number of HStoreFiles on which minor compaction is performed each time. When the size of a file in a Store exceeds the value of this parameter, the file is compacted. You can increase the value of this parameter to reduce the number of times that the file is compacted. If there are too many files in the Store, read performance will be affected.	6

Parameter	Description	Default Value
hbase.hstore.compaction.max	Indicates the maximum number of HStoreFiles on which minor compaction is performed each time. The functions of the parameter and **hbase.hstore.compaction.max.size** are similar. Both are used to limit the execution duration of one compaction.	10
hbase.hstore.compaction.max.size	If the size of an HFile is larger than the parameter value, the HFile will not be compacted in a Minor Compaction but can be compacted in a Major Compaction. The parameter is used to prevent HFiles of large sizes from being compacted. After a Major Compaction is forbidden, multiple HFiles can exist in a Store and will not be merged into one HFile, without affecting data access performance. The unit is byte.	9223372036854775807
hbase.hregion.majorcompaction	Main compression interval of all HStoreFile files in a region. The unit is milliseconds. Execution of Major Compactions consumes much system resources and will affect system performance during peak hours. If service updates, deletion, and reclamation of expired data space are infrequent, set the parameter to **0** to disable Major Compactions. If you must perform a Major Compaction to reclaim more space, increase the parameter value and configure the **hbase.offpeak.end.hour** and **hbase.offpeak.start.hour** parameters to make the Major Compaction be triggered in off-peak hours.	604800000

Parameter	Description	Default Value
• hbase.regionserver.maxlogs • hbase.regionserver.hlog.blocksize	• Indicates the threshold for the number of HLog files that are not flushed on a RegionServer. If the number of HLog files is greater than the threshold, the RegionServer forcibly performs flush operations. • Indicates the maximum size of an HLog file. If the size of an HLog file is greater than the value of this parameter, a new HLog file is generated. The old HLog file is disabled and archived. The two parameters determine the number of HLogs that are not flushed in a RegionServer. When the data volume is less than the total size of memstore, the flush operation is forcibly triggered due to excessive HLog files. In this case, you can adjust the values of the two parameters to avoid forcible flush. Unit: byte	• 32 • 134217728

- **Data writing client tuning**

  It is recommended that data is written in Put List mode if necessary, which greatly improves write performance. The length of each put list needs to be set based on the single put size and parameters of the actual environment. You are advised to do some basic tests before configuring parameters.

- **Data table writing design optimization**

  Table 12-182 Parameters affecting real-time data writing

Parameter	Description	Default Value
COMPRESSION	The compression algorithm compresses blocks in HFiles. For compressible data, configure the compression algorithm to efficiently reduce disk I/Os and improve performance. **NOTE** Some data cannot be efficiently compressed. For example, a compressed figure can hardly be compressed again. The common compression algorithm is SNAPPY, because it has a high encoding/decoding speed and acceptable compression rate.	NONE

Parameter	Description	Default Value
BLOCKSIZE	Different block sizes affect HBase data read and write performance. You can configure sizes for blocks in an HFile. Larger blocks have a higher compression rate. However, they have poor performance in random data read, because HBase reads data in a unit of blocks. Set the parameter to 128 KB or 256 KB to improve data write efficiency without greatly affecting random read performance. The unit is byte.	65536
IN_MEMORY	Whether to cache table data in the memory first, which improves data read performance. If you will frequently access some small tables, set the parameter.	false

12.8.19.5 Improving Real-time Data Read Performance

Scenario

HBase data needs to be read.

Prerequisites

The get or scan interface of HBase has been invoked and data is read in real time from HBase.

Procedure

- **Data reading server tuning**

  Parameter portal:

  Go to the **All Configurations** page of the HBase service. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-183 Configuration items that affect real-time data reading

Parameter	Description	Default Value
GC_OPTS	You can increase HBase memory to improve HBase performance because read and write operations are performed in HBase memory. **HeapSize** and **NewSize** need to be adjusted. When you adjust **HeapSize**, set **Xms** and **Xmx** to the same value to avoid performance problems when JVM dynamically adjusts **HeapSize**. Set **NewSize** to 1/8 of **HeapSize**. ● **HMaster**: If HBase clusters enlarge and the number of Regions grows, properly increase the **GC_OPTS** parameter value of the HMaster. ● **RegionServer**: A RegionServer needs more memory than an HMaster. If sufficient memory is available, increase the **HeapSize** value. **NOTE** When the value of **HeapSize** for the active HMaster is 4 GB, the HBase cluster can support 100,000 regions. Empirically, each time 35,000 regions are added to the cluster, the value of **HeapSize** must be increased by 2 GB. It is recommended that the value of **HeapSize** for the active HMaster not exceed 32 GB.	For versions earlier than MRS 3.x: ● HMaster: -server -Xms2G -Xmx2G -XX:NewSize=256M -XX:MaxNewSize=256M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=512M -XX:MaxDirectMemorySize=512M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps

Parameter	Description	Default Value
		-XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M • RegionServer: -server -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=512M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=512M -XX:MaxDirectMemorySize=512M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcIn

Parameter	Description	Default Value
		terval=0x7FFFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M For MRS 3.*x* or later: ● HMaster -server -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=512M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=512M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -

Parameter	Description	Default Value
		Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M • RegionServer -server -Xms6G -Xmx6G -XX:NewSize=1024M -XX:MaxNewSize=1024M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=512M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSIniti

Parameter	Description	Default Value
		atingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M
hbase.regionserver.handler.count	Indicates the number of requests that RegionServer can process concurrently. If the parameter is set to an excessively large value, threads will compete fiercely. If the parameter is set to an excessively small value, requests will be waiting for a long time in RegionServer, reducing the processing capability. You can add threads based on resources. It is recommended that the value be set to 100 to 300 based on the CPU usage.	200

Parameter	Description	Default Value
hfile.block.cache.size	HBase cache sizes affect query efficiency. Set cache sizes based on query modes and query record distribution. If random query is used to reduce the hit ratio of the buffer, you can reduce the buffer size.	When **offheap** is disabled, the default value is **0.25**. When **offheap** is enabled, the default value is **0.1**.

NOTE

If read and write operations are performed at the same time, the performance of the two operations affects each other. If flush and compaction operations are frequently performed due to data writes, a large number of disk I/O operations are occupied, affecting read performance. If a large number of compaction operations are blocked due to write operations, multiple HFiles exist in the region, affecting read performance. Therefore, if the read performance is unsatisfactory, you need to check whether the write configurations are proper.

- **Data reading client tuning**

  When scanning data, you need to set **caching** (the number of records read from the server at a time. The default value is **1**.). If the default value is used, the read performance will be extremely low.

  If you do not need to read all columns of a piece of data, specify the columns to be read to reduce network I/O.

  If you only need to read the row key, add a filter (FirstKeyOnlyFilter or KeyOnlyFilter) that only reads the row key.

- **Data table reading design optimization**

Table 12-184 Parameters affecting real-time data reading

Parameter	Description	Default Value
COMPRESSION	The compression algorithm compresses blocks in HFiles. For compressible data, configure the compression algorithm to efficiently reduce disk I/Os and improve performance. **NOTE** Some data cannot be efficiently compressed. For example, a compressed figure can hardly be compressed again. The common compression algorithm is SNAPPY, because it has a high encoding/decoding speed and acceptable compression rate.	NONE

Parameter	Description	Default Value
BLOCKSIZE	Different block sizes affect HBase data read and write performance. You can configure sizes for blocks in an HFile. Larger blocks have a higher compression rate. However, they have poor performance in random data read, because HBase reads data in a unit of blocks. Set the parameter to 128 KB or 256 KB to improve data write efficiency without greatly affecting random read performance. The unit is byte.	65536
DATA_BLOCK_ENCODING	Encoding method of the block in an HFile. If a row contains multiple columns, set **FAST_DIFF** to save data storage space and improve performance.	NONE

12.8.19.6 Optimizing JVM Parameters

Scenario

When the number of clusters reaches a certain scale, the default settings of the Java virtual machine (JVM) cannot meet the cluster requirements. In this case, the cluster performance deteriorates or the clusters may be unavailable. Therefore, JVM parameters must be properly configured based on actual service conditions to improve the cluster performance.

Procedure

Navigation path for setting parameters:

The JVM parameters related to the HBase role must be configured in the **hbase-env.sh** file in the **${BIGDATA_HOME}/FusionInsight_HD_*/install/FusionInsight-HBase-2.2.3/hbase/conf/** directory of the node where the HBase service is installed.

Each role has JVM parameter configuration variables, as shown in **Table 12-185**.

Table 12-185 HBase-related JVM parameter configuration variables

Variable	Affected Role
HBASE_OPTS	All roles of HBase
SERVER_GC_OPTS	All roles on the HBase server, such as Master and RegionServer
CLIENT_GC_OPTS	Client process of HBase

Variable	Affected Role
HBASE_MASTER_OPTS	Master of HBase
HBASE_REGIONSERVER_OPTS	RegionServer of HBase
HBASE_THRIFT_OPTS	Thrift of HBase

Configuration example:

```
export HADOOP_NAMENODE_OPTS="-Dhadoop.security.logger=${HADOOP_SECURITY_LOGGER:-INFO,RFAS} -Dhdfs.audit.logger=${HDFS_AUDIT_LOGGER:-INFO,NullAppender} $HADOOP_NAMENODE_OPTS"
```

12.8.20 Common Issues About HBase

12.8.20.1 Why Does a Client Keep Failing to Connect to a Server for a Long Time?

Question

A HBase server is faulty and cannot provide services. In this case, when a table operation is performed on the HBase client, why is the operation suspended and no response is received for a long time?

Answer

Problem Analysis

When the HBase server malfunctions, the table operation request from the HBase client is tried for several times and times out. The default timeout value is **Integer.MAX_VALUE (2147483647 ms)**. The table operation request is retired constantly during such a long period of time and is suspended at last.

Solution

The HBase client provides two configuration items to configure the retry and timeout of the client. **Table 12-186** describes them.

Set the following parameters in the ***Client installation path*/HBase/hbase/conf/hbase-site.xml** configuration file:

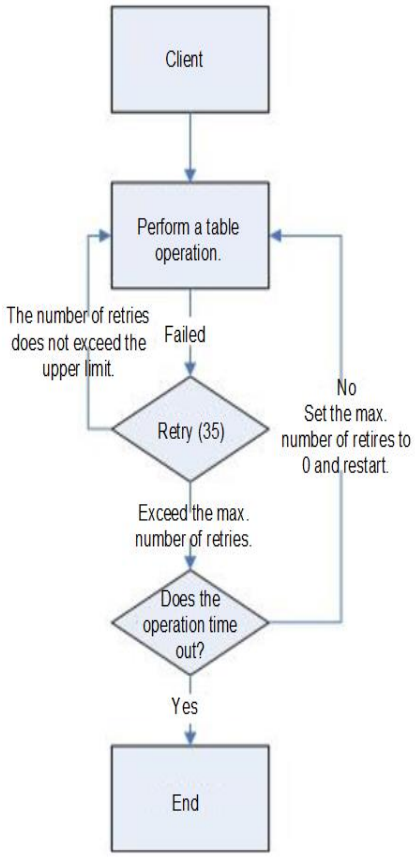
Table 12-186 Configuration parameters of retry and timeout

Parameter	Description	Default Value
hbase.client.operation.timeout	Client operation timeout period You need to manually add the information to the configuration file.	2147483647 ms

Parameter	Description	Default Value
hbase.client.retries.number	Maximum retry times supported by all retryable operations.	35

Figure 12-17 describes the working principles of retry and timeout.

Figure 12-17 Process for HBase client operation retry timeout



The process indicates that a suspension occurs if the preceding parameters are not configured based on site requirements. It is recommended that a proper timeout period be set based on scenarios. If the operation takes a long time, set a long timeout period. If the operation takes a shot time, set a short timeout period. The number of retries can be set to **(hbase.client.retries.number)*60*1000(ms)**. The timeout period can be slightly greater than **hbase.client.operation.timeout**.

12.8.20.2 Operation Failures Occur in Stopping BulkLoad On the Client

Question

Why submitted operations fail by stopping BulkLoad on the client during BulkLoad data importing?

Answer

When BulkLoad is enabled on the client, a partitioner file is generated and used to demarcate the range of Map task data inputting. The file is automatically deleted when BulkLoad exists on the client. In general, if all map tasks are enabled and running, the termination of BulkLoad on the client does not cause the failure of submitted operations. However, due to the retry and speculative execution mechanism of Map tasks, a Map task is performed again if failures of the Reduce task to download the data of the completed Map task exceed the limit. In this case, if BulkLoad already exists on the client, the retry Map task fails and the operation failure occurs because the partitioner file is missing. Therefore, it is recommended not to stop BulkLoad on the client during BulkLoad data importing.

12.8.20.3 Why May a Table Creation Exception Occur When HBase Deletes or Creates the Same Table Consecutively?

Question

When HBase consecutively deletes and creates the same table, why may a table creation exception occur?

Answer

Execution process: Disable Table > Drop Table > Create Table > Disable Table > Drop Table > And more

1. When a table is disabled, HMaster sends an RPC request to RegionServer, and RegionServer brings the region offline. When the time required for closing a region on RegionServer exceeds the timeout period for HBase HMaster to wait for the region to enter the RIT state, HMaster considers that the region is offline by default. Actually, the region may be in the flush memstore phase.
2. After an RPC request is sent to close a region, HMaster checks whether all regions in the table are offline. If the closure times out, HMaster considers that the regions are offline and returns a message indicating that the regions are successfully closed.
3. After the closure is successful, the data directory corresponding to the HBase table is deleted.
4. After the table is deleted, the data directory is recreated by the region that is still in the flush memstore phase.
5. When the table is created again, the **temp** directory is copied to the HBase data directory. However, the HBase data directory is not empty. As a result, when the HDFS rename API is called, the data directory changes to the last layer of the **temp** directory and is appended to the HBase data directory, for example, **$rootDir/data/$nameSpace/$tableName/$tableName**. In this case, the table fails to be created.

Troubleshooting Method

When this problem occurs, check whether the HBase data directory corresponding to the table exists. If it exists, rename the directory.

The HBase data directory consists of **$rootDir/data/$nameSpace/$tableName**, for example, **hdfs://hacluster/hbase/data/default/TestTable**. **$rootDir** is the

HBase root directory, which can be obtained by configuring **hbase.rootdir.perms** in **hbase-site.xml**. The **data** directory is a fixed directory of HBase. **$nameSpace** indicates the nameSpace name. **$tableName** indicates the table name.

12.8.20.4 Why Other Services Become Unstable If HBase Sets up A Large Number of Connections over the Network Port?

Question

Why other services become unstable if HBase sets up a large number of connections over the network port?

Answer

When the OS command ***lsof*** or ***netstat*** is run, it is found that many TCP connections are in the CLOSE_WAIT state and the owner of the connections is HBase RegionServer. This can cause exhaustion of network ports or limit exceeding of HDFS connections, resulting in instability of other services. The HBase CLOSE_WAIT phenomenon is the HBase mechanism.

The reason why HBase CLOSE_WAIT occurs is as follows: HBase data is stored in the HDFS as HFile, which can be called StoreFiles. HBase functions as the client of the HDFS. When HBase creates a StoreFile or starts loading a StoreFile, it creates an HDFS connection. When the StoreFile is created or loaded successfully, the HDFS considers that the task is completed and transfers the connection close permission to HBase. However, HBase may choose not to close the connection to ensure real-time response; that is, HBase may maintain the connection so that it can quickly access the corresponding data file upon request. In this case, the connection is in the CLOSE_WAIT, which indicates that the connection needs to be closed by the client.

When a StoreFile will be created: HBase executes the Flush operation.

When Flush is executed: The data written by HBase is first stored in memstore. The Flush operation is performed only when the usage of memstore reaches the threshold or the ***flush*** command is run to write data into the HDFS.

To resolve the issue, use either of the following methods:

Because of the HBase connection mechanism, the number of StoreFiles must be restricted to reduce the occupation of HBase ports. This can be achieved by triggering HBase's the compaction action, that is, HBase file merging.

Method 1: On HBase shell client, run ***major_compact***.

Method 2: Compile HBase client code to invoke the compact method of the HBaseAdmin class to trigger HBase's compaction action.

If the HBase port occupation issue cannot be resolved through compact, it indicates that the HBase usage has reached the bottleneck. In such a case, you are advised to perform the following:

- Check whether the initial number of Regions configured in the table is appropriate.
- Check whether useless data exists.

If useless data exists, delete the data to reduce the number of storage files for the HBase. If the preceding conditions are not met, then you need to consider a capacity expansion.

12.8.20.5 Why Does the HBase BulkLoad Task (One Table Has 26 TB Data) Consisting of 210,000 Map Tasks and 10,000 Reduce Tasks Fail?

Question

The HBase bulkLoad task (a single table contains 26 TB data) has 210,000 maps and 10,000 reduce tasks (in MRS 3.x or later), and the task fails.

Answer

ZooKeeper I/O bottleneck observation methods:

1. On the monitoring page of Manager, check whether the number of ZooKeeper requests on a single node exceeds the upper limit.
2. View ZooKeeper and HBase logs to check whether a large number of I/O Exception Timeout or SocketTimeout Exception exceptions occur.

Optimization suggestions:

1. Change the number of ZooKeeper instances to 5 or more. You are advised to set **peerType** to **observer** to increase the number of observers.
2. Control the number of concurrent maps of a single task or reduce the memory for running tasks on each node to lighten the node load.
3. Upgrade ZooKeeper data disks, such as SSDs.

12.8.20.6 How Do I Restore a Region in the RIT State for a Long Time?

Question

How do I restore a region in the RIT state for a long time?

Answer

Log in to the HMaster Web UI, choose **Procedure & Locks** in the navigation tree, and check whether any process ID is in the **Waiting** state. If yes, run the following command to release the procedure lock:

hbase hbck -j *Client installation directory*/**HBase/hbase/tools/hbase-hbck2-*.jar bypass -o** *pid*

Check whether the state is in the **Bypass** state. If the procedure on the UI is always in **RUNNABLE(Bypass)** state, perform an active/standby switchover. Run the **assigns** command to bring the region online again.

hbase hbck -j *Client installation directory*/**HBase/hbase/tools/hbase-hbck2-*.jar assigns -o** *regionName*

12.8.20.7 Why Does HMaster Exits Due to Timeout When Waiting for the Namespace Table to Go Online?

Question

Why does HMaster exit due to timeout when waiting for the namespace table to go online?

Answer

During the HMaster active/standby switchover or startup, HMaster performs WAL splitting and region recovery for the RegionServer that failed or was stopped previously.

Multiple threads are running in the background to monitor the HMaster startup process.

- TableNamespaceManager

  This is a help class, which is used to manage the allocation of namespace tables and monitoring table regions during HMaster active/standby switchover or startup. If the namespace table is not online within the specified time (**hbase.master.namespace.init.timeout**, which is 3,600,000 ms by default), the thread terminates HMaster abnormally.

- InitializationMonitor

  This is an initialization thread monitoring class of the primary HMaster, which is used to monitor the initialization of the primary HMaster. If a thread fails to be initialized within the specified time (**hbase.master.initializationmonitor.timeout**, which is 3,600,000 ms by default), the thread terminates HMaster abnormally. If **hbase.master.initializationmonitor.haltontimeout** is started, the default value is **false**.

During the HMaster active/standby switchover or startup, if the **WAL hlog** file exists, the WAL splitting task is initialized. If the WAL hlog splitting task is complete, it initializes the table region allocation task.

HMaster uses ZooKeeper to coordinate log splitting tasks and valid RegionServers and track task development. If the primary HMaster exits during the log splitting task, the new primary HMaster attempts to resend the unfinished task, and RegionServer starts the log splitting task from the beginning.

The initialization of the HMaster is delayed due to the following reasons:

- Network faults occur intermittently.
- Disks run into bottlenecks.
- The log splitting task is overloaded, and RegionServer runs slowly.
- RegionServer (region opening) responds slowly.

In the preceding scenarios, you are advised to add the following configuration parameters to enable HMaster to complete the restoration task earlier. Otherwise, the Master will exit, causing a longer delay of the entire restoration process.

- Increase the online waiting timeout period of the namespace table to ensure that the Master has enough time to coordinate the splitting tasks of the RegionServer worker and avoid repeated tasks.

  hbase.master.namespace.init.timeout (default value: 3,600,000 ms)

- Increase the number of concurrent splitting tasks through RegionServer worker to ensure that RegionServer worker can process splitting tasks in parallel (RegionServers need more cores). Add the following parameters to *Client installation path* **/HBase/hbase/conf/hbase-site.xml**:

  hbase.regionserver.wal.max.splitters (default value: 2)

- If all restoration processes require time, increase the timeout period for initializing the monitoring thread.

  hbase.master.initializationmonitor.timeout (default value: 3,600,000 ms)

12.8.20.8 Why Does SocketTimeoutException Occur When a Client Queries HBase?

Question

Why does the following exception occur on the client when I use the HBase client to operate table data?

```
2015-12-15 02:41:14,054 | WARN  | [task-result-getter-2] | Lost task 2.0 in stage 58.0 (TID 3288, linux-175):
org.apache.hadoop.hbase.client.RetriesExhaustedException: Failed after attempts=36, exceptions:
Tue Dec 15 02:41:14 CST 2015, null, java.net.SocketTimeoutException: callTimeout=60000,
callDuration=60303:
row 'xxxxxx' on table 'xxxxxx' at region=xxxxxx,\x05\x1E
\x80\x00\x00\x00\x80\x00\x00\x00\x00\x00\x00\x00\x80\x00\x00\x00\x00\x00\x000\x00\x80\x00\x00\x0
0\x80\x00\x00\x00\x80\x00\x00,
1449912620868.6a6b7d0c272803d8186930a3bfdb10a9., hostname=xxxxxx,16020,1449941841479,
seqNum=5
at
org.apache.hadoop.hbase.client.RpcRetryingCallerWithReadReplicas.throwEnrichedException(RpcRetryingCall
erWithReadReplicas.java:275)
at org.apache.hadoop.hbase.client.ScannerCallableWithReplicas.call(ScannerCallableWithReplicas.java:223)
at org.apache.hadoop.hbase.client.ScannerCallableWithReplicas.call(ScannerCallableWithReplicas.java:61)
at org.apache.hadoop.hbase.client.RpcRetryingCaller.callWithoutRetries(RpcRetryingCaller.java:200)
at org.apache.hadoop.hbase.client.ClientScanner.call(ClientScanner.java:323)
```

At the same time, the following log is displayed on RegionServer:

```
2015-12-15 02:45:44,551 | WARN  | PriorityRpcServer.handler=7,queue=1,port=16020 | (responseTooSlow):
{"call":"Scan(org.apache.hadoop.hbase.protobuf.generated.ClientProtos$ScanRequest)
","starttimems":1450118730780,"responsesize":416,"method":"Scan","processingtimems":13770,"client":"10.9
1.8.175:41182","queuetimems":0,"class":"HRegionServer"} |
org.apache.hadoop.hbase.ipc.RpcServer.logResponse(RpcServer.java:2221)
2015-12-15 02:45:57,722 | WARN  | PriorityRpcServer.handler=3,queue=1,port=16020 | (responseTooSlow):
{"call":"Scan(org.apache.hadoop.hbase.protobuf.generated.ClientProtos
$ScanRequest)","starttimems":1450118746297,"responsesize":416,
"method":"Scan","processingtimems":11425,"client":"10.91.8.175:41182","queuetimems":1746,"class":"HRegi
onServer"} | org.apache.hadoop.hbase.ipc.RpcServer.logResponse(RpcServer.java:2221)
2015-12-15 02:47:21,668 | INFO  | LruBlockCacheStatsExecutor | totalSize=7.54 GB, freeSize=369.52 MB,
max=7.90 GB, blockCount=406107,
accesses=35400006, hits=16803205, hitRatio=47.47%, , cachingAccesses=31864266, cachingHits=14806045,
cachingHitsRatio=46.47%,
evictions=17654, evicted=16642283, evictedPerRun=942.69189453125 |
org.apache.hadoop.hbase.io.hfile.LruBlockCache.logStats(LruBlockCache.java:858)
2015-12-15 02:52:21,668 | INFO  | LruBlockCacheStatsExecutor | totalSize=7.51 GB, freeSize=395.34 MB,
max=7.90 GB, blockCount=403080,
accesses=35685793, hits=16933684, hitRatio=47.45%, , cachingAccesses=32150053, cachingHits=14936524,
cachingHitsRatio=46.46%,
```

```
evictions=17684, evicted=16800617, evictedPerRun=950.046142578125 |
org.apache.hadoop.hbase.io.hfile.LruBlockCache.logStats(LruBlockCache.java:858)
```

Answer

The memory allocated to RegionServer is too small and the number of Regions is too large. As a result, the memory is insufficient during the running, and the server responds slowly to the client. Modify the following memory allocation parameters in the **hbase-site.xml** configuration file of RegionServer:

Table 12-187 RegionServer memory allocation parameters

Parameter	Description	Default Value
GC_OPTS	Initial memory and maximum memory allocated to RegionServer in startup parameters.	-Xms8G -Xmx8G
hfile.block.cache.size	Percentage of the maximum heap (-Xmx setting) allocated to the block cache of HFiles or StoreFiles.	When **offheap** is disabled, the default value is **0.25**. When **offheap** is enabled, the default value is **0.1**.

12.8.20.9 Why Modified and Deleted Data Can Still Be Queried by Using the Scan Command?

Question

Why modified and deleted data can still be queried by using the **scan** command?

```
scan '<table_name>',{FILTER=>"SingleColumnValueFilter('<column_family>','column',=,'binary:<value>')"}
```

Answer

Because of the scalability of HBase, all values specific to the versions in the queried column are all matched by default, even if the values have been modified or deleted. For a row where column matching has failed (that is, the column does not exist in the row), the HBase also queries the row.

If you want to query only the new values and rows where column matching is successful, you can use the following statement:

```
scan '<table_name>',
{FILTER=>"SingleColumnValueFilter('<column_family>','column',=,'binary:<value>',true,true)"}
```

This command can filter all rows where column query has failed. It queries only the latest values of the current data in the table; that is, it does not query the values before modification or the deleted values.

NOTE

The related parameters of **SingleColumnValueFilter** are described as follows:

SingleColumnValueFilter(final byte[] family, final byte[] qualifier, final CompareOp compareOp, ByteArrayComparable comparator, final boolean filterIfMissing, final boolean latestVersionOnly)

Parameter description:

- family: family of the column to be queried.
- qualifier: column to be queried.
- compareOp: comparison operation, such as = and >.
- comparator: target value to be queried.
- filterIfMissing: whether a row is filtered out if the queried column does not exist. The default value is false.
- latestVersionOnly: whether values of the latest version are queried. The default value is false.

12.8.20.10 Why "java.lang.UnsatisfiedLinkError: Permission denied" exception thrown while starting HBase shell?

Question

Why "java.lang.UnsatisfiedLinkError: Permission denied" exception thrown while starting HBase shell?

Answer

During HBase shell execution JRuby create temporary files under **java.io.tmpdir** path and default value of **java.io.tmpdir** is **/tmp**. If NOEXEC permission is set to /tmp directory then HBase shell start will fail with "java.lang.UnsatisfiedLinkError: Permission denied" exception.

So "java.io.tmpdir" must be set to a different path in HBASE_OPTS/ CLIENT_GC_OPTS if NOEXEC is set to /tmp directory.

12.8.20.11 When does the RegionServers listed under "Dead Region Servers" on HMaster WebUI gets cleared?

Question

When does the RegionServers listed under "Dead Region Servers" on HMaster WebUI gets cleared?

Answer

When an online RegionServer goes down abruptly, it is displayed under "Dead Region Servers" in the HMaster WebUI. When dead RegionServer restarts and reports back to HMaster successfully, the "Dead Region Servers" in the HMaster WebUI gets cleared.

The "Dead Region Servers" is also gets cleared, when the HMaster failover operation is performed successfully.

In cases when an Active HMaster hosting some regions is abruptly killed, Backup HMaster will become the new Active HMater and displays previous Active HMaster as dead RegionServer.

12.8.20.12 Why Are Different Query Results Returned After I Use Same Query Criteria to Query Data Successfully Imported by HBase bulkload?

Question

If the data to be imported by HBase bulkload has identical rowkeys, the data import is successful but identical query criteria produce different query results.

Answer

Data with an identical rowkey is loaded into HBase in the order in which data is read. The data with the latest timestamp is considered to be the latest data. By default, data is not queried by timestamp. Therefore, if you query for data with an identical rowkey, only the latest data is returned.

While data is being loaded by bulkload, the memory processes the data into HFiles quickly, leading to the possibility that data with an identical rowkey has a same timestamp. In this case, identical query criteria may produce different query results.

To avoid this problem, ensure that the same data file does not contain identical rowkeys while you are creating tables or loading data.

12.8.20.13 What Should I Do If I Fail to Create Tables Due to the FAILED_OPEN State of Regions?

Question

What should I do if I fail to create tables due to the FAILED_OPEN state of Regions?

Answer

If a network, HDFS, or Active HMaster fault occurs during the creation of tables, some Regions may fail to go online and therefore enter the FAILED_OPEN state. In this case, tables fail to be created.

The tables that fail to be created due to the preceding mentioned issue cannot be repaired. To solve this problem, perform the following operations to delete and re-create the tables:

1. Run the following command on the cluster client to repair the state of the tables:

   hbase hbck -fixTableStates

2. Enter the HBase shell and run the following commands to delete the tables that fail to be created:

   truncate *'<table_name>'*

disable *'<table_name>'*

drop *'<table_name>'*

3. Create the tables using the recreation command.

12.8.20.14 How Do I Delete Residual Table Names in the /hbase/table-lock Directory of ZooKeeper?

Question

In security mode, names of tables that failed to be created are unnecessarily retained in the table-lock node (default directory is /hbase/table-lock) of ZooKeeper. How do I delete these residual table names?

Answer

Perform the following steps:

1. On the client, run the kinit command as the hbase user to obtain a security certificate.
2. Run the ***hbase zkcli*** command to launch the ZooKeeper Command Line Interface (zkCLI).
3. Run the ***ls /hbase/table*** command on the zkCLI to check whether the table name of the table that fails to be created exists.
   - If the table name exists, no further operation is required.
   - If the table name does not exist, run ***ls /hbase/table-lock*** to check whether the table name of the table fail to be created exist. If the table name exists, run the ***delete /hbase/table-lock/<table>*** command to delete the table name. In the ***delete /hbase/table-lock/<table>*** command, ***<table>*** indicates the residual table name.

12.8.20.15 Why Does HBase Become Faulty When I Set a Quota for the Directory Used by HBase in HDFS?

Question

Why does HBase become faulty when I set quota for the directory used by HBase in HDFS?

Answer

The flush operation of a table is to write memstore data to HDFS.

If the HDFS directory does not have sufficient disk space quota, the flush operation will fail and the region server will stop.

```
Caused by: org.apache.hadoop.hdfs.protocol.DSQuotaExceededException: The DiskSpace quota of /hbase/data/<namespace>/<tableName> is exceeded: quota = 1024 B = 1 KB but diskspace consumed = 402655638 B = 384.00 MB
?at org.apache.hadoop.hdfs.server.namenode.DirectoryWithQuotaFeature.verifyStoragespaceQuota(DirectoryWithQuotaFeature.java:211)
?at org.apache.hadoop.hdfs.server.namenode.DirectoryWithQuotaFeature.verifyQuota(DirectoryWithQuotaFeatu
```

```
re.java:239)
?at org.apache.hadoop.hdfs.server.namenode.FSDirectory.verifyQuota(FSDirectory.java:882)
?at org.apache.hadoop.hdfs.server.namenode.FSDirectory.updateCount(FSDirectory.java:711)
?at org.apache.hadoop.hdfs.server.namenode.FSDirectory.updateCount(FSDirectory.java:670)
?at org.apache.hadoop.hdfs.server.namenode.FSDirectory.addBlock(FSDirectory.java:495)
```

In the preceding exception, the disk space quota of the **/hbase/data/<namespace>/<tableName>** table is 1 KB, but the memstore data is 384.00 MB. Therefore, the flush operation fails and the region server stops.

When the region server is terminated, HMaster replays the WAL file of the terminated region server to restore data. The disk space quota is limited. As a result, the replay operation of the WAL file fails, and the HMaster process exits unexpectedly.

```
2016-07-28 19:11:40,352 | FATAL | MASTER_SERVER_OPERATIONS-10-91-9-131:16000-0 | Caught throwable
while processing event M_SERVER_SHUTDOWN |
org.apache.hadoop.hbase.master.HMaster.abort(HMaster.java:2474)
java.io.IOException: failed log splitting for 10-91-9-131,16020,1469689987884, will retry
?at
org.apache.hadoop.hbase.master.handler.ServerShutdownHandler.resubmit(ServerShutdownHandler.java:365
)
?at
org.apache.hadoop.hbase.master.handler.ServerShutdownHandler.process(ServerShutdownHandler.java:220)
?at org.apache.hadoop.hbase.executor.EventHandler.run(EventHandler.java:129)
?at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
?at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
?at java.lang.Thread.run(Thread.java:745)
Caused by: java.io.IOException: error or interrupted while splitting logs in [hdfs://hacluster/hbase/WALs/<RS-
Hostname>,<RS-Port>,<startcode>-splitting] Task = installed = 6 done = 3 error = 3
?at org.apache.hadoop.hbase.master.SplitLogManager.splitLogDistributed(SplitLogManager.java:290)
?at org.apache.hadoop.hbase.master.MasterFileSystem.splitLog(MasterFileSystem.java:402)
?at org.apache.hadoop.hbase.master.MasterFileSystem.splitLog(MasterFileSystem.java:375)
```

Therefore, you cannot set the quota value for the HBase directory in HDFS. If the exception occurs, perform the following operations:

Step 1 Run the **kinit** *Username* command on the client to enable the HBase user to obtain security authentication.

Step 2 Run the **hdfs dfs -count -q** */hbase/data/<namespace>/<tableName>* command to check the allocated disk space quota.

Step 3 Run the following command to cancel the quota limit and restore HBase:

hdfs dfsadmin -clrSpaceQuota */hbase/data/<namespace>/<tableName>*

----End

12.8.20.16 Why HMaster Times Out While Waiting for Namespace Table to be Assigned After Rebuilding Meta Using OfflineMetaRepair Tool and Startups Failed

Question

Why HMaster times out while waiting for namespace table to be assigned after rebuilding meta using OfflineMetaRepair tool and startups failed?

HMaster abort with following FATAL message,

```
2017-06-15 15:11:07,582 FATAL [Hostname:16000.activeMasterManager] master.HMaster: Unhandled
exception. Starting shutdown.
```

```
java.io.IOException: Timedout 120000ms waiting for namespace table to be assigned
    at org.apache.hadoop.hbase.master.TableNamespaceManager.start(TableNamespaceManager.java:98)
    at org.apache.hadoop.hbase.master.HMaster.initNamespace(HMaster.java:1054)
    at org.apache.hadoop.hbase.master.HMaster.finishActiveMasterInitialization(HMaster.java:848)
    at org.apache.hadoop.hbase.master.HMaster.access$600(HMaster.java:199)
    at org.apache.hadoop.hbase.master.HMaster$2.run(HMaster.java:1871)
    at java.lang.Thread.run(Thread.java:745)
```

Answer

When meta is rebuilt by OfflineMetaRepair tool then HMaster wait for all region server's WAL split during start up to avoid the data inconsistency problem. HMaster trigger user regions assignment once WAL split completes. So when the cluster is in the unusual scenario, there are chances WAL splitting may take long time which depends on multiple factors like too many WALs, slow I/O, region servers are not stable etc.

HMaster should be able to finish all region server WAL splitting successfully. Perform the following steps.

1. Make sure cluster is stable, no other problem exist. If any problem occurs, please correct them first.
2. Configure a large value to **hbase.master.initializationmonitor.timeout** parameters, default value is **3600000** milliseconds.
3. Restart HBase service.

12.8.20.17 Why Messages Containing FileNotFoundException and no lease Are Frequently Displayed in the HMaster Logs During the WAL Splitting Process?

Question

Why messages containing FileNotFoundException and no lease are frequently displayed in the HMaster logs during the WAL splitting process?

```
2017-06-10 09:50:27,586 | ERROR | split-log-closeStream-2 | Couldn't close log at hdfs://hacluster/hbase/
data/default/largeT1/2b48346d087275fe751fc049334fda93/recovered.edits/0000000000000000000.temp |
org.apache.hadoop.hbase.wal.WALSplitter$LogRecoveredEditsOutputSink$2.call(WALSplitter.java:1330)
java.io.FileNotFoundException: No lease on /hbase/data/default/
largeT1/2b48346d087275fe751fc049334fda93/recovered.edits/0000000000000000000.temp (inode
1092653): File does not exist. [Lease.  Holder: DFSClient_NONMAPREDUCE_1202985678_1, pendingcreates:
1936]
?at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkLease(FSNamesystem.java:3432)
?at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.analyzeFileState(FSNamesystem.java:3223)
?at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.getNewBlockTargets(FSNamesystem.java:3057)
?at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.getAdditionalBlock(FSNamesystem.java:3011)
?at org.apache.hadoop.hdfs.server.namenode.NameNodeRpcServer.addBlock(NameNodeRpcServer.java:842)
?at
org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolServerSideTranslatorPB.addBlock(ClientNameno
deProtocolServerSideTranslatorPB.java:526)
?at org.apache.hadoop.hdfs.protocol.proto.ClientNamenodeProtocolProtos$ClientNamenodeProtocol
$2.callBlockingMethod(ClientNamenodeProtocolProtos.java)
?at org.apache.hadoop.ipc.ProtobufRpcEngine$Server$ProtoBufRpcInvoker.call(ProtobufRpcEngine.java:616)
?at org.apache.hadoop.ipc.RPC$Server.call(RPC.java:973)
?at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2260)
?at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2256)
?at java.security.AccessController.doPrivileged(Native Method)
?at javax.security.auth.Subject.doAs(Subject.java:422)
?at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1769)
?at org.apache.hadoop.ipc.Server$Handler.run(Server.java:2254)
```

```
?at sun.reflect.GeneratedConstructorAccessor40.newInstance(Unknown Source)
?at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
?at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
?at org.apache.hadoop.ipc.RemoteException.instantiateException(RemoteException.java:106)
?at org.apache.hadoop.ipc.RemoteException.unwrapRemoteException(RemoteException.java:73)
?at org.apache.hadoop.hdfs.DataStreamer.locateFollowingBlock(DataStreamer.java:1842)
?at org.apache.hadoop.hdfs.DataStreamer.nextBlockOutputStream(DataStreamer.java:1639)
?at org.apache.hadoop.hdfs.DataStreamer.run(DataStreamer.java:665)
```

Answer

During the WAL splitting process, the WAL splitting timeout period is specified by the **hbase.splitlog.manager.timeout** parameter. If the WAL splitting process fails to complete within the timeout period, the task is submitted again. Multiple WAL splitting tasks may be submitted during a specified period. If the **temp** file is deleted when one WAL splitting task completes, other tasks cannot find the file and the FileNotFoudException exception is reported. To avoid the problem, perform the following modifications:

The default value of **hbase.splitlog.manager.timeout** is 600,000 ms. The cluster specification is that each RegionServer has 2,000 to 3,000 regions. When the cluster is normal (HBase is normal and HDFS does not have a large number of read and write operations), you are advised to adjust this parameter based on the cluster specifications. If the actual specifications (the actual average number of regions on each RegionServer) are greater than the default specifications (the default average number of regions on each RegionServer, that is, 2,000), the adjustment solution is (actual specifications/default specifications) x Default time.

Set the **splitlog** parameter in the **hbase-site.xml** file on the server. **Table 12-188** describes the parameter.

Table 12-188 Description of the **splitlog** parameter

Parameter	Description	Default Value
hbase.splitlog.manager.timeout	Timeout period for receiving worker response by the distributed SplitLog management program.	600000

12.8.20.18 Why Does the ImportTsv Tool Display "Permission denied" When the Same Linux User as and a Different Kerberos User from the Region Server Are Used?

Question

When the same Linux user (for example, user **omm**) as and a different Kerberos user (for example, user **admin**) from the Region Server are used, why does the ImportTsv tool fail to be executed and the error message "Permission denied" is displayed?

```
Exception in thread "main" org.apache.hadoop.security.AccessControlException: Permission denied:
user=admin, access=WRITE, inode="/user/omm-bulkload/hbase-staging/
partitions_cab16de5-87c2-4153-9cca-a6f4ed4278a6":hbase:hadoop:drwx--x--x
        at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.check(FSPermissionChecker.java:342)
```

```
        at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.check(FSPermissionChecker.java:315)
        at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:23
1)
        at
com.xxx.hadoop.adapter.hdfs.plugin.HWAccessControlEnforce.checkPermission(HWAccessControlEnforce.java:
69)
        at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:19
0)
        at org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPermission(FSDirectory.java:1789)
        at org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPermission(FSDirectory.java:1773)
        at org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkAncestorAccess(FSDirectory.java:1756)
        at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.startFileInternal(FSNamesystem.java:2490)
        at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.startFileInt(FSNamesystem.java:2425)
        at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.startFile(FSNamesystem.java:2308)
        at
org.apache.hadoop.hdfs.server.namenode.NameNodeRpcServer.create(NameNodeRpcServer.java:745)
        at
org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolServerSideTranslatorPB.create(ClientNamenodeP
rotocolServerSideTranslatorPB.java:434)
        at org.apache.hadoop.hdfs.protocol.proto.ClientNamenodeProtocolProtos$ClientNamenodeProtocol
$2.callBlockingMethod(ClientNamenodeProtocolProtos.java)
        at org.apache.hadoop.ipc.ProtobufRpcEngine$Server
$ProtoBufRpcInvoker.call(ProtobufRpcEngine.java:616)
        at org.apache.hadoop.ipc.RPC$Server.call(RPC.java:973)
        at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2260)
        at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2256)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1781)
        at org.apache.hadoop.ipc.Server$Handler.run(Server.java:2254)
```

Answer

The ImportTsv tool creates a partition file in the HBase temporary directory specified by **hbase.fs.tmp.dir** in the *Client installation path* **/HBase/hbase/conf/hbase-site.xml file.** Therefore, the client (Kerberos user) must have the **rwx** permission on the specified temporary directory to perform the ImportTsv operation. The default value of **hbase.fs.tmp.dir** is **/user/${user.name}/hbase-staging** (for example, **/user/omm/hbase-staging**). **$ {user.name}** indicates the OS username (user **omm**). The client (Kerberos user, for example, user **admin**) does not have the **rwx** permission on the directory.

To solve the preceding problem, perform the following steps:

1. On the client, set **hbase.fs.tmp.dir** to the directory of the current Kerberos user (for example, **/user/admin/hbase-staging**), or provide the **rwx** permission required by the configured directory for the client (Kerberos user).
2. Perform the ImportTsv operation again.

12.8.20.19 Insufficient Rights When a Tenant Accesses Phoenix

Question

When a tenant accesses Phoenix, a message is displayed indicating that the tenant has insufficient rights.

Answer

You need to associate the HBase service and Yarn queues when creating a tenant.

The tenant must be granted additional rights to perform operations on Phoenix, that is, the RWX permission on the Phoenix system table.

Example:

Tenant **hbase** has been created. Log in to the HBase Shell as user **admin** and run the **scan 'hbase:acl'** command to query the role of the tenant. The role is **hbase_1450761169920** (in the format of tenant name_timestamp).

Run the following commands to grant rights to the tenant (if the Phoenix system table has not been generated, log in to the Phoenix client as user **admin** first and then grant rights on the HBase Shell):

grant '@hbase_1450761169920','RWX','SYSTEM.CATALOG'

grant '@hbase_1450761169920','RWX','SYSTEM.FUNCTION'

grant '@hbase_1450761169920','RWX','SYSTEM.SEQUENCE'

grant '@hbase_1450761169920','RWX','SYSTEM.STATS'

Create user **phoenix** and bind it with tenant **hbase**, so that tenant **hbase** can access the Phoenix client as user **phoenix**.

12.8.20.20 What Can I Do When HBase Fails to Recover a Task and a Message Is Displayed Stating "Rollback recovery failed"?

Question

The system automatically rolls back data after an HBase recovery task fails. If "Rollback recovery failed" is displayed, the rollback fails. After the rollback fails, data stops being processed and the junk data may be generated. How can I resolve this problem?

Answer

You need to manually clear the junk data before performing the backup or recovery task next time.

Step 1 Install the cluster client in **/opt/client**.

Step 2 Run **source /opt/client/bigdata_env** as the client installation user to configure the environment variable.

Step 3 Run **kinit admin** for administrator authentication.

Step 4 Run **zkCli.sh -server** *business IP address of ZooKeeper*:**2181** to connect to the ZooKeeper.

Step 5 Run **deleteall /recovering** to delete the junk data. Run **quit** to disconnect ZooKeeper.

NOTE

Running this command will cause data loss. Exercise caution.

Step 6 Run **hdfs dfs -rm -f -r /user/hbase/backup** to delete temporary data.

Step 7 Log in to FusionInsight Manager and choose **O&M**. In the navigation pane on the left, choose **Backup and Restoration** > **Restoration Management**. In the task list, locate the row that contains the target task and click **View History** in the **Operation** column. In the displayed dialog box, click ⌄ before a specified execution record to view the snapshot name.

Snapshot [*snapshot name*] is created successfully before recovery.

Step 8 Switch to the client, run **hbase shell**, and then **delete_all_snapshot** '*snapshot name*.*' to delete the temporary snapshot.

----End

12.8.20.21 How Do I Fix Region Overlapping?

Question

When the HBaseFsck tool is used to check the region status in MRS 3.x and later versions, if the log contains **ERROR: (regions region1 and region2) There is an overlap in the region chain** or **ERROR: (region region1) Multiple regions have the same startkey: xxx**, overlapping exists in some regions. How do I solve this problem?

Answer

To rectify the fault, perform the following steps:

Step 1 Run the **hbase hbck -repair** *tableName* command to restore the table that contains overlapping.

Step 2 Run the **hbase hbck** *tableName* command to check whether overlapping exists in the restored table.

- If overlapping does not exist, go to **Step 3**.
- If overlapping exists, go to **Step 1**.

Step 3 Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **More** > **Perform HMaster Switchover** to complete the HMaster active/standby switchover.

Step 4 Run the **hbase hbck** *tableName* command to check whether overlapping exists in the restored table.

- If overlapping does not exist, no further action is required.
- If overlapping still exists, start from **Step 1** to perform the recovery again.

----End

12.8.20.22 Why Does RegionServer Fail to Be Started When GC Parameters Xms and Xmx of HBase RegionServer Are Set to 31 GB?

Question

(MRS 3.x and later versions) Check the **hbase-omm-*.out** log of the node where RegionServer fails to be started. It is found that the log contains **An error report file with more information is saved as: /tmp/hs_err_pid*.log**. Check the **/tmp/**

hs_err_pid*.log file. It is found that the log contains **#Internal Error (vtableStubs_aarch64.cpp:213), pid=9456, tid=0x0000ffff97fdd200 and #guarantee(__ pc() <= s->code_end()) failed: overflowed buffer**, indicating that the problem is caused by JDK. How do I solve this problem?

Answer

To rectify the fault, perform the following steps:

Step 1 Run the **su - omm** command on a node where RegionServer fails to be started to switch to user **omm**.

Step 2 Run the **java -XX:+PrintFlagsFinal -version |grep HeapBase** command as user **omm**. Information similar to the following is displayed:

```
uintx HeapBaseMinAddress = 2147483648 {pd product}
```

Step 3 Change the values of **-Xms** and **-Xmx** in **GC_OPTS** to values that are not between **32G-HeapBaseMinAddress** and **32G**, excluding the values of **32G** and **32G-HeapBaseMinAddress**.

Step 4 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HBase** > **Instance**, select the failed instance, and choose **More** > **Restart Instance** to restart the failed instance.

----End

12.8.20.23 Why Does the LoadIncrementalHFiles Tool Fail to Be Executed and "Permission denied" Is Displayed When Nodes in a Cluster Are Used to Import Data in Batches?

Question

Why does the LoadIncrementalHFiles tool fail to be executed and "Permission denied" is displayed when a Linux user is manually created in a normal cluster and DataNode in the cluster is used to import data in batches?

```
2020-09-20 14:53:53,808 WARN  [main] shortcircuit.DomainSocketFactory: error creating DomainSocket
java.net.ConnectException: connect(2) error: Permission denied when trying to connect to '/var/run/
FusionInsight-HDFS/dn_socket'
    at org.apache.hadoop.net.unix.DomainSocket.connect0(Native Method)
    at org.apache.hadoop.net.unix.DomainSocket.connect(DomainSocket.java:256)
    at org.apache.hadoop.hdfs.shortcircuit.DomainSocketFactory.createSocket(DomainSocketFactory.java:168)
    at org.apache.hadoop.hdfs.client.impl.BlockReaderFactory.nextDomainPeer(BlockReaderFactory.java:804)
    at
org.apache.hadoop.hdfs.client.impl.BlockReaderFactory.createShortCircuitReplicaInfo(BlockReaderFactory.java
:526)
    at org.apache.hadoop.hdfs.shortcircuit.ShortCircuitCache.create(ShortCircuitCache.java:785)
    at org.apache.hadoop.hdfs.shortcircuit.ShortCircuitCache.fetchOrCreate(ShortCircuitCache.java:722)
    at
org.apache.hadoop.hdfs.client.impl.BlockReaderFactory.getBlockReaderLocal(BlockReaderFactory.java:483)
    at org.apache.hadoop.hdfs.client.impl.BlockReaderFactory.build(BlockReaderFactory.java:360)
    at org.apache.hadoop.hdfs.DFSInputStream.getBlockReader(DFSInputStream.java:663)
    at org.apache.hadoop.hdfs.DFSInputStream.blockSeekTo(DFSInputStream.java:594)
    at org.apache.hadoop.hdfs.DFSInputStream.readWithStrategy(DFSInputStream.java:776)
    at org.apache.hadoop.hdfs.DFSInputStream.read(DFSInputStream.java:845)
    at java.io.DataInputStream.readFully(DataInputStream.java:195)
    at org.apache.hadoop.hbase.io.hfile.FixedFileTrailer.readFromStream(FixedFileTrailer.java:401)
    at org.apache.hadoop.hbase.io.hfile.HFile.isHFileFormat(HFile.java:651)
    at org.apache.hadoop.hbase.io.hfile.HFile.isHFileFormat(HFile.java:634)
    at org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.visitBulkHFiles(LoadIncrementalHFiles.java:1090)
```

```
    at
org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.discoverLoadQueue(LoadIncrementalHFiles.java:1006)
    at
org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.prepareHFileQueue(LoadIncrementalHFiles.java:257)
    at org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.doBulkLoad(LoadIncrementalHFiles.java:364)
    at org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.run(LoadIncrementalHFiles.java:1263)
    at org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.run(LoadIncrementalHFiles.java:1276)
    at org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.run(LoadIncrementalHFiles.java:1311)
    at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:76)
    at org.apache.hadoop.hbase.tool.LoadIncrementalHFiles.main(LoadIncrementalHFiles.java:1333)
```

Answer

If the client that the LoadIncrementalHFiles tool depends on is installed in the cluster and is on the same node as DataNode, HDFS creates short-circuit read during the execution of the tool to improve performance. The short-circuit read depends on the **/var/run/FusionInsight-HDFS** directory (**dfs.domain.socket.path**). The default permission on this directory is **750**. This user does not have the permission to operate the directory.

To solve the preceding problem, perform the following operations:

Method 1: Create a user (recommended).

Step 1 Create a user on Manager. By default, the user group contains the **ficommon** group.

```
[root@xxx-xxx-xxx-xxx ~]# id test
uid=20038(test) gid=9998(ficommon) groups=9998(ficommon)
```

Step 2 Import data again.

----End

Method 2: Change the owner group of the current user.

Step 1 Add the user to the **ficommon** group.

```
[root@xxx-xxx-xxx-xxx ~]# usermod -a -G ficommon test
[root@xxx-xxx-xxx-xxx ~]# id test
uid=2102(test) gid=2102(test) groups=2102(test),9998(ficommon)
```

Step 2 Import data again.

----End

12.8.20.24 Why Is the Error Message "import argparse" Displayed When the Phoenix sqlline Script Is Used?

Question

When the sqlline script is used on the client, the error message "import argparse" is displayed.

Answer

Step 1 Log in to the node where the HBase client is installed as user **root**. Perform security authentication using the **hbase** user.

Step 2 Go to the directory where the sqlline script of the HBase client is stored and run the **python3 sqlline.py** command.

----End

12.8.20.25 How Do I Deal with the Restrictions of the Phoenix BulkLoad Tool?

Question

When the indexed field data is updated, if a batch of data exists in the user table, the BulkLoad tool cannot update the global and partial mutable indexes.

Answer

Problem Analysis

1. Create a table.

```
CREATE TABLE TEST_TABLE(
DATE varchar not null,
NUM integer not null,
SEQ_NUM integer not null,
ACCOUNT1 varchar not null,
ACCOUNTDES varchar,
FLAG varchar,
SALL double,
CONSTRAINT PK PRIMARY KEY (DATE,NUM,SEQ_NUM,ACCOUNT1)
);
```

2. Create a global index.

   CREATE INDEX TEST_TABLE_INDEX ON TEST_TABLE(ACCOUNT1,DATE,NUM,ACCOUNTDES,SEQ_NUM);

3. Insert data.

   UPSERT INTO TEST_TABLE (DATE,NUM,SEQ_NUM,ACCOUNT1,ACCOUNTDES,FLAG,SALL) values ('20201001',30201001,13,'367392332','sffa1','','');

4. Execute the BulkLoad task to update data.

   hbase org.apache.phoenix.mapreduce.CsvBulkLoadTool -t TEST_TABLE -i /tmp/test.csv, where the content of **test.csv** is as follows:

20201001	30201001	13	367392332	sffa888	1231243	23

5. Symptom: The existing index data cannot be directly updated. As a result, two pieces of index data exist.

```
+------------+-----------+-----------+---------------+----------------+
| :ACCOUNT1  |  :DATE    |  :NUM     | 0:ACCOUNTDES  | :SEQ_NUM  |
+------------+-----------+-----------+---------------+----------------+
| 367392332  | 20201001  | 30201001  | sffa1         | 13             |
| 367392332  | 20201001  | 30201001  | sffa888       | 13             |
+------------+-----------+-----------+---------------+----------------+
```

Solution

Step 1 Delete the old index table.

DROP INDEX TEST_TABLE_INDEX ON TEST_TABLE;

Step 2 Create an index table in asynchronous mode.

CREATE INDEX TEST_TABLE_INDEX ON TEST_TABLE(ACCOUNT1,DATE,NUM,ACCOUNTDES,SEQ_NUM) ASYNC;

Step 3 Recreate a index.

hbase org.apache.phoenix.mapreduce.index.IndexTool --data-table TEST_TABLE --index-table TEST_TABLE_INDEX --output-path /user/test_table

----End

12.8.20.26 Why a Message Is Displayed Indicating that the Permission is Insufficient When CTBase Connects to the Ranger Plug-ins?

Question

When CTBase accesses the HBase service with the Ranger plug-ins enabled and you are creating a cluster table, a message is displayed indicating that the permission is insufficient.

```
ERROR: Create ClusterTable failed. Error: org.apache.hadoop.hbase.security.AccessDeniedException:
Insufficient permissions for user 'ctbase2@HADOOP.COM' (action=create)
at org.apache.ranger.authorization.hbase.AuthorizationSession.publishResults(AuthorizationSession.java:278)
at
org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor.authorizeAccess(RangerAuthorizatio
nCoprocessor.java:654)
at
org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor.requirePermission(RangerAuthorizati
onCoprocessor.java:772)
at
org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor.preCreateTable(RangerAuthorization
Coprocessor.java:943)
at
org.apache.ranger.authorization.hbase.RangerAuthorizationCoprocessor.preCreateTable(RangerAuthorization
Coprocessor.java:428)
at org.apache.hadoop.hbase.master.MasterCoprocessorHost$12.call(MasterCoprocessorHost.java:351)
at org.apache.hadoop.hbase.master.MasterCoprocessorHost$12.call(MasterCoprocessorHost.java:348)
at org.apache.hadoop.hbase.coprocessor.CoprocessorHost
$ObserverOperationWithoutResult.callObserver(CoprocessorHost.java:581)
at org.apache.hadoop.hbase.coprocessor.CoprocessorHost.execOperation(CoprocessorHost.java:655)
at
org.apache.hadoop.hbase.master.MasterCoprocessorHost.preCreateTable(MasterCoprocessorHost.java:348)
at org.apache.hadoop.hbase.master.HMaster$5.run(HMaster.java:2192)
at
org.apache.hadoop.hbase.master.procedure.MasterProcedureUtil.submitProcedure(MasterProcedureUtil.java:1
34)
at org.apache.hadoop.hbase.master.HMaster.createTable(HMaster.java:2189)
at org.apache.hadoop.hbase.master.MasterRpcServices.createTable(MasterRpcServices.java:711)
at org.apache.hadoop.hbase.shaded.protobuf.generated.MasterProtos$MasterService
$2.callBlockingMethod(MasterProtos.java)
at org.apache.hadoop.hbase.ipc.RpcServer.call(RpcServer.java:458)
at org.apache.hadoop.hbase.ipc.CallRunner.run(CallRunner.java:133)
at org.apache.hadoop.hbase.ipc.RpcExecutor$Handler.run(RpcExecutor.java:338)
at org.apache.hadoop.hbase.ipc.RpcExecutor$Handler.run(RpcExecutor.java:318)
```

Answer

CTBase users can configure permission policies on the Ranger page and grant the READ, WRITE, CREATE, ADMIN, and EXECUTE permissions to the CTBase metadata table **_ctmeta_**, cluster table, and index table.

12.9 Using HDFS

12.9.1 Using Hadoop from Scratch

You can use Hadoop to submit wordcount jobs. Wordcount is the most classic Hadoop job and is used to count the number of words in massive text.

Procedure

Step 1 Prepare the wordcount program.

Multiple open source Hadoop sample programs are provided, including wordcount. You can download the Hadoop sample program from **https://dist.apache.org/repos/dist/release/hadoop/common/**.

For example, select hadoop-2.10.x, download **hadoop-2.10.x.tar.gz**, decompress it, and obtain **hadoop-2.10.x\share\hadoop\mapreduce** (the Hadoop sample program) from **hadoop-mapreduce-examples-2.10.x.jar**. The **hadoop-mapreduce-examples-2.10.x.jar** sample program contains the wordcount program.

NOTE

hadoop-2.10.x indicates the Hadoop version.

Step 2 Prepare data files.

There is no format requirement for data files. Prepare one or more **.txt** files. The following are examples of the **.txt** file:

```
qwsdfhoedfrffrofhuncckgktpmhutopmma
jjpsffjfjorgjgtyiuyjmhombmbogohoyhm
jhheyeombdhuaqqiquyebchdhmamdhdemmj
doeyhjwedcrfvtgbmojiyhhqssddddddfkf
kjhhjkehdeiyrudjhfhfhffooqweopuyyyy
```

Step 3 Upload data to OBS.

1. Log in to OBS Console.
2. Click **Parallel File System** and choose **Create Parallel File System** to create a file system named **wordcount01**.

   wordcount01 is only an example. The file system name must be globally unique. Otherwise, the parallel file system fails to be created.
3. In the OBS file system list, click **wordcount01** and choose **Files** > **Create Folder** to create the **program** and **input** folders.
   - **program**: stores user programs.
   - **input**: stores user data files.
4. Go to the **program** folder, choose **Upload File** > **add file**, select the program package downloaded in **Step 1** from the local host, and click **Upload**.
5. Go to the **input** folder and upload the data file prepared in **Step 2** to the **input** folder.

Step 4 Log in to the MRS console. In the navigation pane on the left, click **Clusters** and choose **Active Clusters**. Click the cluster name. The cluster must contain Hadoop components.

Step 5 Submit the wordcount job.

On the MRS console, click the **Jobs** tab and click **Create**. The **Create Job** page is displayed.

- Set **Type** to **MapReduce**.
- Set **Name** to **mr_01**.
- Set the path of the executable program to the address of the program stored on the OBS. For example: **obs://wordcount01/program/hadoop-mapreduce-examples-2.10.x.jar**
- Enter **wordcount obs://wordcount01/input/ obs://wordcount01/output/** in the **Parameter** pane.

  NOTE

  - Replace the OBS file system name in **obs://wordcount01/input/** with the actual name of the file system created in the environment.
  - Replace the OBS file system name in **obs://wordcount01/output/** with the actual name of the file system created in the environment. Enter a directory that does not exist in the **output** directory.
- **Service Parameter** can be left blank.

A job can be submitted only when the cluster is in the **Running** state.

After a job is submitted successfully, it is in the **Accepted** state by default. You do not need to manually execute the job.

Step 6 View the job execution result.

1. Go to the **Jobs** tab page and check whether the job is successfully executed.

   It takes some time to run the job. After the job is complete, refresh the job list

   Once a job has succeeded or failed, you cannot execute it again. However, you can add or copy a job, and set job parameters to submit a job again.
2. Log in to the OBS console, go to the OBS path, and view the job output information.

   You can view output files in the **output** directory created in **Step 5**. You need to download the file to the local host and open it in text format.

----End

12.9.2 Configuring Memory Management

Scenario

In HDFS, each file object needs to register corresponding information in the NameNode and occupies certain storage space. As the number of files increases, if the original memory space cannot store the corresponding information, you need to change the memory size.

Configuration Description

Navigation path for setting parameters:

Go to the **All Configurations** page of HDFS by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-189 Parameter description

Parameter	Description	Default Value
GC_PROFILE	The NameNode memory size depends on the size of FsImage, which can be calculated based on the following formula: FsImage size = Number of files x 900 bytes. You can estimate the memory size of the NameNode of HDFS based on the calculation result. The value range of this parameter is as follows: • **high**: 4 GB • **medium**: 2 GB • **low**: 256 MB • **custom**: The memory size can be set according to the data size in GC_OPTS.	custom

Parameter	Description	Default Value
GC_OPTS	JVM parameter used for garbage collection (GC). This parameter is valid only when **GC_PROFILE** is set to **custom**. Ensure that the **GC_OPT** parameter is set correctly. Otherwise, the process will fail to be started. **NOTICE** Exercise caution when you modify the configuration. If the configuration is incorrect, the services are unavailable.	-Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=128M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -Djdk.tls.ephemeralDHKeySize=2048

12.9.3 Creating an HDFS Role

Scenario

This section describes how to create and configure an HDFS role on FusionInsight Manager. The HDFS role is granted the rights to read, write, and execute HDFS directories or files.

A user has the complete permission on the created HDFS directories or files, that is, the user can directly read data from and write data to as well as authorize others to access the HDFS directories or files.

NOTE

- This section applies to MRS 3.*x* or later.
- An HDFS role can be created only in security mode.
- If the current component uses Ranger for permission control, HDFS policies must be configured based on Ranger for permission management. For details, see **Adding a Ranger Access Permission Policy for HDFS**.

Prerequisites

The system administrator has understood the service requirements.

Procedure

Step 1 Log in to FusionInsight Manager, and choose **System** > **Permission** > **Role**.

Step 2 On the displayed page, click **Create Role** and fill in **Role Name** and **Description**.

Step 3 Configure the resource permission. For details, see **Table 12-190**.

File System: HDFS directory and file permission

Common HDFS directories are as follows:

- **flume**: Flume data storage directory
- **hbase**: HBase data storage directory
- **mr-history**: MapReduce task information storage directory
- **tmp**: temporary data storage directory
- **user**: user data storage directory

Table 12-190 Setting a role

Task	Operation
Setting the HDFS administrator permission	In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS, and select **Cluster Admin Operations**. **NOTE** The setting takes effect after the HDFS service is restarted.
Setting the permission for users to check and recover HDFS	1. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS > **File System**. 2. Locate the save path of specified directories or files on HDFS. 3. In the **Permission** column of the specified directories or files, select **Read** and **Execute**.
Setting the permission for users to read directories or files of other users	1. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS > **File System**. 2. Locate the save path of specified directories or files on HDFS. 3. In the **Permission** column of the specified directories or files, select **Read** and **Execute**.

Task	Operation
Setting the permission for users to write data to files of other users	1. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS > **File System**. 2. Locate the save path of specified files on HDFS. 3. In the **Permission** column of the specified files, select **Write** and **Execute**.
Setting the permission for users to create or delete sub-files or sub-directories in the directory of other users	1. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS > **File System**. 2. Locate the path where the specified directory is saved in the HDFS. 3. In the **Permission** column of the specified directories, select **Write** and **Execute**.
Setting the permission for users to execute directories or files of other users	1. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS > **File System**. 2. Locate the save path of specified directories or files on HDFS. 3. In the **Permission** column of the specified directories or files, select **Execute**.
Setting the permission for allowing subdirectories to inherit all permissions of their parent directories	1. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > HDFS > **File System**. 2. Locate the save path of specified directories or files on HDFS. 3. In the **Permission** column of the specified directories or files, select **Recursive**.

Step 4 Click **OK**, and return to the **Role** page.

----End

12.9.4 Using the HDFS Client

Scenario

This section describes how to use the HDFS client in an O&M scenario or service scenario.

Prerequisites

- The client has been installed.

  For example, the installation directory is **/opt/hadoopclient**. The client directory in the following operations is only an example. Change it to the actual installation directory.

- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user needs to change the password upon the first login. (This operation is not required in normal mode.)

Using the HDFS Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the HDFS Shell command. Example:

hdfs dfs -ls /

----End

Common HDFS Client Commands

The following table lists common HDFS client commands.

For more commands, see **https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/CommandsManual.html#User_Commands**.

Table 12-191 Common HDFS client commands

Command	Description	Example
hdfs dfs -mkdir *Folder name*	Used to create a folder.	**hdfs dfs -mkdir /tmp/mydir**
hdfs dfs -ls *Folder name*	Used to view a folder.	**hdfs dfs -ls /tmp**
hdfs dfs -put *Local file on the client node Specified HDFS path*	Used to upload a local file to a specified HDFS path.	**hdfs dfs -put /opt/test.txt /tmp** Upload the **/opt/test.txt** file on the client node to the **/tmp** directory of HDFS.
hdfs dfs -get *Specified file on HDFS Specified path on the client node*	Used to download the HDFS file to the specified local path.	**hdfs dfs -get /tmp/test.txt /opt/** Download the **/tmp/test.txt** file on HDFS to the **/opt** path on the client node.

Command	Description	Example
hdfs dfs -rm -r -f *Specified folder on HDFS*	Used to delete a folder.	**hdfs dfs -rm -r -f /tmp/mydir**
hdfs dfs -chmod *Permission parameter File directory*	Used to configure the HDFS directory permission for a user.	**hdfs dfs -chmod 700 /tmp/test**

Client-related FAQs

1. What do I do when the HDFS client exits abnormally and error message "java.lang.OutOfMemoryError" is displayed after the HDFS client command is running?

   This problem occurs because the memory required for running the HDFS client exceeds the preset upper limit (128 MB by default). You can change the memory upper limit of the client by modifying **CLIENT_GC_OPTS** in *<Client installation path>***/HDFS/component_env**. For example, if you want to set the upper limit to 1 GB, run the following command:

   ```
   CLIENT_GC_OPTS="-Xmx1G"
   ```

   After the modification, run the following command to make the modification take effect:

   source <*Client installation path*>/**/bigdata_env**

2. How do I set the log level when the HDFS client is running?

   By default, the logs generated during the running of the HDFS client are printed to the console. The default log level is INFO. To enable the DEBUG log level for fault locating, run the following command to export an environment variable:

   export HADOOP_ROOT_LOGGER=DEBUG,console

   Then run the HDFS Shell command to generate the DEBUG logs.

   If you want to print INFO logs again, run the following command:

   export HADOOP_ROOT_LOGGER=INFO,console

3. How do I delete HDFS files permanently?

   HDFS provides a recycle bin mechanism. Typically, after an HDFS file is deleted, the file is moved to the recycle bin of HDFS. If the file is no longer needed and the storage space needs to be released, clear the corresponding recycle bin directory, for example, **hdfs://hacluster/user/xxx/.Trash/Current/***xxx*.

12.9.5 Running the DistCp Command

Scenario

DistCp is a tool used to perform large-amount data replication between clusters or in a cluster. It uses MapReduce tasks to implement distributed copy of a large amount of data.

Prerequisites

- The Yarn client or a client that contains Yarn has been installed. For example, the installation directory is **/opt/client**.
- Service users of each component are created by the system administrator based on service requirements. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login. (Not involved in normal mode)
- To copy data between clusters, you need to enable the inter-cluster data copy function on both clusters.

Procedure

Step 1 Log in to the node where the client is installed.

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, the user group to which the user executing the DistCp command belongs must be **supergroup** and the user run the following command to perform user authentication. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the DistCp command. The following provides an example:

hadoop distcp hdfs://hacluster/source hdfs://hacluster/target

----End

Common Usage of DistCp

1. The following is an example of the commonest usage of DistCp:
   ```
   hadoop distcp -numListstatusThreads 40 -update -delete -prbugpaxtq hdfs://cluster1/source hdfs://cluster2/target
   ```

   NOTE

   In the preceding command:

   - **-numListstatusThreads** specifies the number of threads for creating the list of 40 copied files.
   - **-update -delete** specifies that files at the source location and the target location are synchronized, and that files with excessive target locations are deleted. If you need to copy files incrementally, delete **-delete**.
   - If **-prbugpaxtq** and **-update** are used, it indicates that the status information of the copied file is also updated.
   - **hdfs://cluster1/source** indicates the source location, and **hdfs://cluster2/target** indicates the target location.

2. The following is an example of data copy between clusters:
   ```
   hadoop distcp hdfs://cluster1/foo/bar hdfs://cluster2/bar/foo
   ```

NOTE

The network between cluster1 and cluster2 must be reachable, and the two clusters must use the same HDFS version or compatible HDFS versions.

3. The following are multiple examples of data copy in a source directory:

```
hadoop distcp hdfs://cluster1/foo/a \
hdfs://cluster1/foo/b \
hdfs://cluster2/bar/foo
```

The preceding command is used to copy the folders a and b of cluster1 to the **/bar/foo** directory of cluster2. The effect is equivalent to that of the following commands:

```
hadoop distcp -f hdfs://cluster1/srclist \
hdfs://cluster2/bar/foo
```

The content of **srclist** is as follows. Before running the DistCp command, upload the **srclist** file to HDFS.

```
hdfs://cluster1/foo/a
hdfs://cluster1/foo/b
```

4. **-update** indicates that a to-be-copied file does not exist in the target location, or the content of the copied file in the target location is updated; and **-overwrite** is used to overwrite existing files in the target location.

The following is an example of the difference between no option and any one of the two options (either **update** or **overwrite**) that is added:

Assume that the structure of a file at the source location is as follows:

```
hdfs://cluster1/source/first/1
hdfs://cluster1/source/first/2
hdfs://cluster1/source/second/10
hdfs://cluster1/source/second/20
```

Commands without options are as follows:

```
hadoop distcp hdfs://cluster1/source/first  hdfs://cluster1/source/second  hdfs://cluster2/target
```

By default, the preceding command creates the **first** and **second** folders at the target location. Therefore, the copy results are as follows:

```
hdfs://cluster2/target/first/1
hdfs://cluster2/target/first/2
hdfs://cluster2/target/second/10
hdfs://cluster2/target/second/20
```

The command with any one of the two options (for example, **update**) is as follows:

```
hadoop distcp -update hdfs://cluster1/source/first  hdfs://cluster1/source/second  hdfs://cluster2/target
```

The preceding command copies only the content at the source location to the target location. Therefore, the copy results are as follows:

```
hdfs://cluster2/target/1
hdfs://cluster2/target/2
hdfs://cluster2/target/10
hdfs://cluster2/target/20
```

NOTE

- If files with the same name exist in multiple source locations, the DistCp command fails.
- If neither **update** nor **overwrite** is used and the file to be copied already exists in the target location, the file will be skipped.
- When **update** is used, if the file to be copied already exists in the target location but the file content is different, the file content in the target location is updated.
- When **overwrite** is used, if the file to be copied already exists in the target location, the file in the target location is still overwritten.

5. The following table describes other command options:

Table 12-192 Other command options

Option	Description
-p[rbugpcaxtq]	When **-update** is also used, the status information of a copied file is updated even if the content of the copied file is not updated. **r**: number of copies **b**: size of a block **u**: user to which the files belong **g**: user group to which the user belongs **p**: permission **c**: check and type **a**: access control **t**: timestamp **q**: quota information
-i	Failures ignored during copying
-log <logdir>	Path of the specified log
-v	Additional information in the specified log
-m <num_maps>	Maximum number of concurrent copy tasks that can be executed at the same time
-numListstatusThreads	Number of threads for constituting the list of copied files. This option increases the running speed of DistCp.
-overwrite	File at the target location that is to be overwritten
-update	A file at the target location is updated if the size and check of a file at the source location are different from those of the file at the target location.
-append	When **-update** is also used, the content of the file at the source location is added to the file at the target location.

Option	Description
-f <urilist_uri>	Content of the **<urilist_uri>** file is used as the file list to be copied.
-filters	A local file is specified whose content contains multiple regular expressions. If the file to be copied matches a regular expression, the file is not copied.
-async	The **distcp** command is run asynchronously.
-atomic {-tmp <tmp_dir>}	An atomic copy can be performed. You can add a temporary directory during copying.
-bandwidth	The transmission bandwidth of each copy task. Unit: MB/s.
-delete	The files that exist in the target location is deleted but do not exist in the source location. This option is usually used with **-update**, and indicates that files at the source location are synchronized with those at the target location and the redundant files at the target location are deleted.
-diff <oldSnapshot> <newSnapshot>	The differences between the old and new versions are copied to a file in the old version at the target location.
-skipcrccheck	Whether to skip the cyclic redundancy check (CRC) between the source file and the target file.
-strategy {dynamic| uniformsize}	The policy for copying a task. The default policy is **uniformsize**, that is, each copy task copies the same number of bytes.

FAQs of DistCp

1. When you run the DistCp command, if the content of some copied files is large, you are advised to change the timeout period of MapReduce that executes the copy task. It can be implemented by specifying the **mapreduce.task.timeout** in the DistCp command. For example, run the following command to change the timeout to 30 minutes:

   ```
   hadoop distcp -Dmapreduce.task.timeout=1800000 hdfs://cluster1/source hdfs://cluster2/target
   ```

   Or, you can also use **filters** to exclude the large files out of the copy process. The command example is as follows:

   ```
   hadoop distcp -filters /opt/client/filterfile hdfs://cluster1/source hdfs://cluster2/target
   ```

   In the preceding command, *filterfile* indicates a local file, which contains multiple expressions used to match the path of a file that is not copied. The following is an example:

   ```
   .*excludeFile1.*
   .*excludeFile2.*
   ```

2. If the DistCp command unexpectedly quits, the error message "java.lang.OutOfMemoryError" is displayed.

This is because the memory required for running the copy command exceeds the preset memory limit (default value: 128 MB). You can change the memory upper limit of the client by modifying **CLIENT_GC_OPTS** in *<Client installation path>***/HDFS/component_env**. For example, if you want to set the memory upper limit to 1 GB, refer to the following configuration:

```
CLIENT_GC_OPTS="-Xmx1G"
```

After the modification, run the following command to make the modification take effect:

source {*Client installation path*}**/bigdata_env**

3. When the dynamic policy is used to run the DistCp command, the command exits unexpectedly and the error message "Too many chunks created with splitRatio" is displayed.

   The cause of this problem is that the value of **distcp.dynamic.max.chunks.tolerable** (default value: 20,000) is less than the value of **distcp.dynamic.split.ratio** (default value: 2) multiplied by the number of Maps. This problem occurs when the number of Maps exceeds 10,000. You can use the **-m** parameter to reduce the number of Maps to less than 10,000.

   ```
   hadoop distcp -strategy dynamic -m 9500 hdfs://cluster1/source hdfs://cluster2/target
   ```

   Alternatively, you can use the **-D** parameter to set **distcp.dynamic.max.chunks.tolerable** to a large value.

   ```
   hadoop distcp -Ddistcp.dynamic.max.chunks.tolerable=30000 -strategy dynamic hdfs://cluster1/source hdfs://cluster2/target
   ```

12.9.6 Overview of HDFS File System Directories

This section describes the directory structure in HDFS, as shown in the following table.

Table 12-193 HDFS directory structure (applicable to versions earlier than MRS 3.x)

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/tmp/spark/sparkhive-scratch	Fixed directory	Stores temporary files of metastore sessions in Spark JDBCServer.	No	Failed to run the task.
/tmp/sparkhive-scratch	Fixed directory	Stores temporary files of metastore session that are executed using Spark CLI.	No	Failed to run the task.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/tmp/carbon/	Fixed directory	Stores the abnormal data in this directory if abnormal CarbonData data exists during data import.	Yes	Error data is lost.
/tmp/Loader-${*Job name*}_${*MR job ID*}	Temporary directory	Stores the region information about Loader HBase bulkload jobs. The data is automatically deleted after the job running is completed.	No	Failed to run the Loader HBase Bulkload job.
/tmp/logs	Fixed directory	Stores the collected MR task logs.	Yes	MR task logs are lost.
/tmp/archived	Fixed directory	Archives the MR task logs on HDFS.	Yes	MR task logs are lost.
/tmp/hadoop-yarn/staging	Fixed directory	Stores the run logs, summary information, and configuration attributes of ApplicationMaster running jobs.	No	Services are running improperly.
/tmp/hadoop-yarn/staging/history/done_intermediate	Fixed directory	Stores temporary files in the **/tmp/hadoop-yarn/staging** directory after all tasks are executed.	No	MR task logs are lost.
/tmp/hadoop-yarn/staging/history/done	Fixed directory	The periodic scanning thread periodically moves the **done_intermediate** log file to the **done** directory.	No	MR task logs are lost.
/tmp/mr-history	Fixed directory	Stores the historical record files that are pre-loaded.	No	Historical MR task log data is lost.
/tmp/hive	Fixed directory	Stores Hive temporary files.	No	Failed to run the Hive task.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/tmp/hive-scratch	Fixed directory	Stores temporary data (such as session information) generated during Hive running.	No	Failed to run the current task.
/user/{user}/.sparkStaging	Fixed directory	Stores temporary files of the SparkJDBCServer application.	No	Failed to start the executor.
/user/spark/jars	Fixed directory	Stores running dependency packages of the Spark executor.	No	Failed to start the executor.
/user/loader	Fixed directory	Stores dirty data of Loader jobs and data of HBase jobs.	No	Failed to execute the HBase job. Or dirty data is lost.
/user/loader/etl_dirty_data_dir				
/user/loader/etl_hbase_putlist_tmp				
/user/loader/etl_hbase_tmp				
/user/mapred	Fixed directory	Stores Hadoop-related files.	No	Failed to start Yarn.
/user/hive	Fixed directory	Stores Hive-related data by default, including the depended Spark lib package and default table data storage path.	No	User data is lost.
/user/omm-bulkload	Temporary directory	Stores HBase batch import tools temporarily.	No	Failed to import HBase tasks in batches.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/user/hbase	Temporary directory	Stores HBase batch import tools temporarily.	No	Failed to import HBase tasks in batches.
/sparkJobHistory	Fixed directory	Stores Spark event log data.	No	The History Server service is unavailable, and the task fails to be executed.
/flume	Fixed directory	Stores data collected by Flume from HDFS.	No	Flume runs improperly.
/mr-history/tmp	Fixed directory	Stores logs generated by MapReduce jobs.	Yes	Log information is lost.
/mr-history/done	Fixed directory	Stores logs managed by MR JobHistory Server.	Yes	Log information is lost.
/tenant	Created when a tenant is added.	Directory of a tenant in the HDFS. By default, the system automatically creates a folder in the **/tenant** directory based on the tenant name. For example, the default HDFS storage directory for **ta1** is **tenant/ta1**. When a tenant is created for the first time, the system creates the **/tenant** directory in the HDFS root directory. You can customize the storage path.	No	The tenant account is unavailable.
/apps{1~5}/	Fixed directory	Stores the Hive package used by WebHCat.	No	Failed to run the WebHCat tasks.
/hbase	Fixed directory	Stores HBase data.	No	HBase user data is lost.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/hbaseFileStream	Fixed directory	Stores HFS files.	No	The HFS file is lost and cannot be restored.
/ats/active	Fixed directory	HDFS path used to store the timeline data of running applications.	No	Failed to run the **tez** task after the directory deletion.
/ats/done	Fixed directory	HDFS path used to store the timeline data of completed applications.	No	Automatically created after the deletion.
/flink	Fixed directory	Stores the checkpoint task data.	No	Failed to run tasks after the deletion.

Table 12-194 Directory structure of the HDFS file system (applicable to MRS 3.x or later)

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/tmp/spark2x/sparkhive-scratch	Fixed directory	Stores temporary files of metastore session in Spark2x JDBCServer.	No	Failed to run the task.
/tmp/sparkhive-scratch	Fixed directory	Stores temporary files of metastore sessions that are executed in CLI mode using Spark2x CLI.	No	Failed to run the task.
/tmp/logs/	Fixed directory	Stores container log files.	Yes	Container log files cannot be viewed.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/tmp/carbon/	Fixed directory	Stores the abnormal data in this directory if abnormal CarbonData data exists during data import.	Yes	Error data is lost.
/tmp/Loader-${*Job name*}_${*MR job ID*}	Temporary directory	Stores the region information about Loader HBase bulkload jobs. The data is automatically deleted after the job running is completed.	No	Failed to run the Loader HBase Bulkload job.
/tmp/hadoop-omm/yarn/system/rmstore	Fixed directory	Stores the ResourceManager running information.	Yes	Status information is lost after ResourceManager is restarted.
/tmp/archived	Fixed directory	Archives the MR task logs on HDFS.	Yes	MR task logs are lost.
/tmp/hadoop-yarn/staging	Fixed directory	Stores the run logs, summary information, and configuration attributes of ApplicationMaster running jobs.	No	Services are running improperly.
/tmp/hadoop-yarn/staging/history/done_intermediate	Fixed directory	Stores temporary files in the **/tmp/hadoop-yarn/staging** directory after all tasks are executed.	No	MR task logs are lost.
/tmp/hadoop-yarn/staging/history/done	Fixed directory	The periodic scanning thread periodically moves the **done_intermediate** log file to the **done** directory.	No	MR task logs are lost.
/tmp/mr-history	Fixed directory	Stores the historical record files that are pre-loaded.	No	Historical MR task log data is lost.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/tmp/hive-scratch	Fixed directory	Stores temporary data (such as session information) generated during Hive running.	No	Failed to run the current task.
/user/{user}/.sparkStaging	Fixed directory	Stores temporary files of the SparkJDBCServer application.	No	Failed to start the executor.
/user/spark2x/jars	Fixed directory	Stores running dependency packages of the Spark2x executor.	No	Failed to start the executor.
/user/loader	Fixed directory	Stores dirty data of Loader jobs and data of HBase jobs.	No	Failed to execute the HBase job. Or dirty data is lost.
/user/loader/etl_dirty_data_dir				
/user/loader/etl_hbase_putlist_tmp				
/user/loader/etl_hbase_tmp				
/user/oozie	Fixed directory	Stores dependent libraries required for Oozie running, which needs to be manually uploaded.	No	Failed to schedule Oozie.
/user/mapred/hadoop-mapreduce-*3.1.1*.tar.gz	Fixed files	Stores JAR files used by the distributed MR cache.	No	The MR distributed cache function is unavailable.
/user/hive	Fixed directory	Stores Hive-related data by default, including the depended Spark lib package and default table data storage path.	No	User data is lost.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/user/omm-bulkload	Temporary directory	Stores HBase batch import tools temporarily.	No	Failed to import HBase tasks in batches.
/user/hbase	Temporary directory	Stores HBase batch import tools temporarily.	No	Failed to import HBase tasks in batches.
/spark2xJobHistory2x	Fixed directory	Stores Spark2x eventlog data.	No	The History Server service is unavailable, and the task fails to be executed.
/flume	Fixed directory	Stores data collected by Flume from HDFS.	No	Flume runs improperly.
/mr-history/tmp	Fixed directory	Stores logs generated by MapReduce jobs.	Yes	Log information is lost.
/mr-history/done	Fixed directory	Stores logs managed by MR JobHistory Server.	Yes	Log information is lost.
/tenant	Created when a tenant is added.	Directory of a tenant in the HDFS. By default, the system automatically creates a folder in the **/tenant** directory based on the tenant name. For example, the default HDFS storage directory for **ta1** is **tenant/ta1**. When a tenant is created for the first time, the system creates the **/tenant** directory in the HDFS root directory. You can customize the storage path.	No	The tenant account is unavailable.

Path	Type	Function	Whether the Directory Can Be Deleted	Deletion Consequence
/apps{1~5}/	Fixed directory	Stores the Hive package used by WebHCat.	No	Failed to run the WebHCat tasks.
/hbase	Fixed directory	Stores HBase data.	No	HBase user data is lost.
/hbaseFileStream	Fixed directory	Stores HFS files.	No	The HFS file is lost and cannot be restored.

12.9.7 Changing the DataNode Storage Directory

Scenario

NOTE

This section applies to MRS 3.*x* or later.

If the storage directory defined by the HDFS DataNode is incorrect or the HDFS storage plan changes, the system administrator needs to modify the DataNode storage directory on FusionInsight Manager to ensure that the HDFS works properly. Changing the ZooKeeper storage directory includes the following scenarios:

- Change the storage directory of the DataNode role. In this way, the storage directories of all DataNode instances are changed.
- Change the storage directory of a single DataNode instance. In this way, only the storage directory of this instance is changed, and the storage directories of other instances remain the same.

Impact on the System

- The HDFS service needs to be stopped and restarted during the process of changing the storage directory of the DataNode role, and the cluster cannot provide services before it is completely started.
- The DataNode instance needs to stopped and restarted during the process of changing the storage directory of the instance, and the instance at this node cannot provide services before it is started.
- The directory for storing service parameter configurations must also be updated.

Prerequisites

- New disks have been prepared and installed on each data node, and the disks are formatted.
- New directories have been planned for storing data in the original directories.
- The HDFS client has been installed.
- The system administrator user **hdfs** is available.
- When changing the storage directory of a single DataNode instance, ensure that the number of active DataNode instances is greater than the value of **dfs.replication**.

Procedure

Check the environment.

Step 1 Log in to the server where the HDFS client is installed as user **root**, and run the following command to configure environment variables:

source *Installation directory of the HDFS client*/**bigdata_env**

Step 2 If the cluster is in security mode, run the following command to authenticate the user:

kinit hdfs

Step 3 Run the following command on the HDFS client to check whether all directories and files in the HDFS root directory are normal:

hdfs fsck /

Check the fsck command output.

- If the following information is displayed, no file is lost or damaged. Go to **Step 4**.
  ```
  The filesystem under path '/' is HEALTHY
  ```
- If other information is displayed, some files are lost or damaged. Go to **Step 5**.

Step 4 Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services**, and check whether **Running Status** of HDFS is **Normal**.

- If yes, go to **Step 6**.
- If no, the HDFS status is unhealthy. Go to **Step 5**.

Step 5 Rectify the HDFS fault.. The task is complete.

Step 6 Determine whether to change the storage directory of the DataNode role or that of a single DataNode instance:

- To change the storage directory of the DataNode role, go to **Step 7**.
- To change the storage directory of a single DataNode instance, go to **Step 12**.

Changing the storage directory of the DataNode role

Step 7 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Stop Instance** to stop the HDFS service.

Step 8 Log in to each data node where the HDFS service is installed as user **root** and perform the following operations:

1. Create a target directory (**data1** and **data2** are original directories in the cluster).

   For example, to create a target directory **${BIGDATA_DATA_HOME}/hadoop/data3/dn**, run the following command:

   mkdir -p ${BIGDATA_DATA_HOME}/hadoop/data3/dn

2. Mount the target directory to the new disk. For example, mount **${BIGDATA_DATA_HOME}/hadoop/data3** to the new disk.
3. Modify permissions on the new directory.

   For example, to create a target directory **${BIGDATA_DATA_HOME}/hadoop/data3/dn**, run the following commands:

   chmod 700 ${BIGDATA_DATA_HOME}/hadoop/data3/dn -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/hadoop/data3/dn -R**

4. Copy the data to the target directory.

   For example, if the old directory is **${BIGDATA_DATA_HOME}/hadoop/data1/dn** and the target directory is **${BIGDATA_DATA_HOME}/hadoop/data3/dn**, run the following command:

   cp -af ${BIGDATA_DATA_HOME}/hadoop/data1/dn/* ${BIGDATA_DATA_HOME}/hadoop/data3/dn

Step 9 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations** > **All Configurations** to go to the HDFS service configuration page.

Change the value of **dfs.datanode.data.dir** from the default value **%{@auto.detect.datapart.dn}** to the new target directory, for example, **${BIGDATA_DATA_HOME}/hadoop/data3/dn**.

For example, the original data storage directories are **/srv/BigData/hadoop/data1**, **/srv/BigData/hadoop/data2**. To migrate data from the **/srv/BigData/hadoop/data1** directory to the newly created **/srv/BigData/hadoop/data3** directory, replace the whole parameter with **/srv/BigData/hadoop/data2, /srv/BigData/hadoop/data3**. Separate multiple storage directories with commas (,). In this example, changed directories are **/srv/BigData/hadoop/data2**, **/srv/BigData/hadoop/data3**.

Step 10 Click **Save**. Choose **Cluster** > *Name of the desired cluster* > **Services**. On the page that is displayed, start the services that have been stopped.

Step 11 After the HDFS is started, run the following command on the HDFS client to check whether all directories and files in the HDFS root directory are correctly copied:

hdfs fsck /

Check the fsck command output.

- If the following information is displayed, no file is lost or damaged, and data replication is successful. No further action is required.

  ```
  The filesystem under path '/' is HEALTHY
  ```

- If other information is displayed, some files are lost or damaged. In this case, check whether **8.4** is correct and run the **hdfs fsck** *Name of the damaged file* **-delete** command.

Changing the storage directory of a single DataNode instance

Step 12 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Instance**. Select the HDFS instance whose storage directory needs to be modified, and choose **More** > **Stop Instance**.

Step 13 Log in to the DataNode node as user **root**, and perform the following operations:

1. Create a target directory.

   For example, to create a target directory **${BIGDATA_DATA_HOME}/hadoop/data3/dn**, run the following command:

   mkdir -p ${BIGDATA_DATA_HOME}/hadoop/data3/dn

2. Mount the target directory to the new disk.

   For example, mount **${BIGDATA_DATA_HOME}/hadoop/data3** to the new disk.

3. Modify permissions on the new directory.

   For example, to create a target directory **${BIGDATA_DATA_HOME}/hadoop/data3/dn**, run the following commands:

   chmod 700 ${BIGDATA_DATA_HOME}/hadoop/data3/dn -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/hadoop/data3/dn -R**

4. Copy the data to the target directory.

   For example, if the old directory is **${BIGDATA_DATA_HOME}/hadoop/data1/dn** and the target directory is **${BIGDATA_DATA_HOME}/hadoop/data3/dn**, run the following command:

   cp -af ${BIGDATA_DATA_HOME}/hadoop/data1/dn/* ${BIGDATA_DATA_HOME}/hadoop/data3/dn

Step 14 On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **HDFS** > **Instance**. Click the specified DataNode instance and go to the **Configurations** page.

Change the value of **dfs.datanode.data.dir** from the default value **%{@auto.detect.datapart.dn}** to the new target directory, for example, **${BIGDATA_DATA_HOME}/hadoop/data3/dn**.

For example, the original data storage directories are **/srv/BigData/hadoop/data1,/srv/BigData/hadoop/data2**. To migrate data from the **/srv/BigData/hadoop/data1** directory to the newly created **/srv/BigData/hadoop/data3** directory, replace the whole parameter with **/srv/BigData/hadoop/data2,/srv/BigData/hadoop/data3**.

Step 15 Click **Save**, and then click **OK**.

Operation succeeded is displayed. click **Finish**.

Step 16 Choose **More** > **Restart Instance** to restart the DataNode instance.

----End

12.9.8 Configuring HDFS Directory Permission

Scenario

The permission for some HDFS directories is **777** or **750** by default, which brings potential security risks. You are advised to modify the permission for the HDFS directories after the HDFS is installed to increase user security.

Procedure

Log in to the HDFS client as the administrator and run the following command to modify the permission for the **/user** directory.

The permission is set to **1777**, that is, **1** is added to the original permission. This indicates that only the user who creates the directory can delete it.

hdfs dfs -chmod 1777 */user*

To ensure security of the system file, you are advised to harden the security for non-temporary directories. The following directories are examples:

- /user:777
- /mr-history:777
- /mr-history/tmp:777
- /mr-history/done:777
- /user/mapred:755

12.9.9 Configuring NFS

Scenario

NOTE

This section applies to MRS 3.*x* or later.

Before deploying a cluster, you can deploy a Network File System (NFS) server based on requirements to store NameNode metadata to enhance data reliability.

If the NFS server has been deployed and NFS services are configured, you can follow operations in this section to configure NFS on the cluster. These operations are optional.

Procedure

Step 1 Check the permission of the shared NFS directories on the NFS server to ensure that the server can access NameNode in the MRS cluster.

Step 2 Log in to the active NameNode as user **root**.

Step 3 Run the following commands to create a directory and assign it write permissions:

mkdir ${BIGDATA_DATA_HOME}/namenode-nfs

chown omm:wheel ${BIGDATA_DATA_HOME}/namenode-nfs

chmod 750 ${BIGDATA_DATA_HOME}/namenode-nfs

Step 4 Run the following command to mount the NFS to the active NameNode:

mount -t nfs -o rsize=8192,wsize=8192,soft,nolock,timeo=3,intr *IP address of the NFS server*:*Shared directory* **${BIGDATA_DATA_HOME}/namenode-nfs**

For example, if the IP address of the NFS server is **192.168.0.11** and the shared directory is **/opt/Hadoop/NameNode**, run the following command:

mount -t nfs -o rsize=8192,wsize=8192,soft,nolock,timeo=3,intr 192.168.0.11:/opt/Hadoop/NameNode ${BIGDATA_DATA_HOME}/namenode-nfs

Step 5 Perform **Step 2** to **Step 4** on the standby NameNode.

NOTE

The names of the shared directories (for example, **/opt/Hadoop/NameNode**) created on the NFS server by the active and standby NameNodes must be different.

Step 6 Log in to FusionInsight Manager, and choose **Cluster** > *Name of the desired cluster* > **Service** > **HDFS** > **Configuration** > **All Configurations**.

Step 7 In the search box, search for **dfs.namenode.name.dir**, add **${BIGDATA_DATA_HOME}/namenode-nfs** to **Value**, and click **Save**. Separate paths with commas (,).

Step 8 Click **OK**. On the **Dashboard** tab page, choose **More** > **Restart Service** to restart the service.

----End

12.9.10 Planning HDFS Capacity

In HDFS, DataNode stores user files and directories as blocks, and file objects are generated on the NameNode to map each file, directory, and block on the DataNode.

The file objects on the NameNode require certain memory capacity. The memory consumption linearly increases as more file objects generated. The number of file objects on the NameNode increases and the objects consume more memory when the files and directories stored on the DataNode increase. In this case, the existing hardware may not meet the service requirement and the cluster is difficult to be scaled out.

Capacity planning of the HDFS that stores a large number of files is to plan the capacity specifications of the NameNode and DataNode and to set parameters according to the capacity plans.

Capacity Specifications

- NameNode capacity specifications

  Each file object on the NameNode corresponds to a file, directory, or block on the DataNode.

  A file uses at least one block. The default size of a block is **134,217,728**, that is, 128 MB, which can be set in the **dfs.blocksize** parameter. By default, a file whose size is less than 128 MB occupies only one block. If the file size is greater than 128 MB, the number of occupied blocks is the file size divided by

128 MB (Number of occupied blocks = File size/128). The directories do not occupy any blocks.

Based on **dfs.blocksize**, the number of file objects on the NameNode is calculated as follows:

Table 12-195 Number of NameNode file objects

Size of a File	Number of File Objects
< 128 MB	1 (File) + 1 (Block) = 2
> 128 MB (for example, 128 GB)	1 (File) + 1,024 (128 GB/128 MB = 1,024 blocks) = 1,025

The maximum number of file objects supported by the active and standby NameNodes is 300,000,000 (equivalent to 150,000,000 small files). **dfs.namenode.max.objects** specifies the number of file objects that can be generated in the system. The default value is **0**, which indicates that the number of generated file objects is not limited.

- DataNode capacity specifications

  In HDFS, blocks are stored on the DataNode as copies. The default number of copies is **3**, which can be set in the **dfs.replication** parameter.

  The number of blocks stored on all DataNode role instances in the cluster can be calculated based on the following formula: Number of HDFS blocks x 3 Average number of saved blocks = Number of HDFS blocks x 3/Number of DataNodes

  Table 12-196 DataNode specifications

Item	Specifications
Maximum number of blocks supported by a DataNode instance	5,000,000
Maximum number of blocks supported by a disk on a DataNode instance	500,000
Minimum number of disks required when the number of blocks supported by a DataNode instance reaches the maximum	10

  Table 12-197 Number of DataNodes

Number of HDFS Blocks	Minimum Number of DataNode Roles
10,000,000	10,000,000 *3/5,000,000 = 6
50,000,000	50,000,000 *3/5,000,000 = 30

Number of HDFS Blocks	Minimum Number of DataNode Roles
100,000,000	100,000,000 *3/5,000,000 = 60

Setting Memory Parameters

- Configuration rules of the NameNode JVM parameter

  Default value of the NameNode JVM parameter **GC_OPTS**:

  -Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=128M -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -Djdk.tls.ephemeralDHKeySize=3072 -Djdk.tls.rejectClientInitiatedRenegotiation=true -Djava.io.tmpdir=${Bigdata_tmp_dir}

  The number of NameNode files is proportional to the used memory size of the NameNode. When file objects change, you need to change **-Xms2G -Xmx4G -XX:NewSize=128M --XX:MaxNewSize=256M** in the default value. The following table lists the reference values.

  Table 12-198 NameNode JVM configuration

Number of File Objects	Reference Value
10,000,000	-Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
20,000,000	-Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G
50,000,000	-Xms32G -Xmx32G -XX:NewSize=3G -XX:MaxNewSize=3G
100,000,000	-Xms64G -Xmx64G -XX:NewSize=6G -XX:MaxNewSize=6G
200,000,000	-Xms96G -Xmx96G -XX:NewSize=9G -XX:MaxNewSize=9G
300,000,000	-Xms164G -Xmx164G -XX:NewSize=12G -XX:MaxNewSize=12G

- Configuration rules of the DataNode JVM parameter

  Default value of the DataNode JVM parameter **GC_OPTS**:

  -Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M -XX:MetaspaceSize=128M -XX:MaxMetaspaceSize=128M -

XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:CMSInitiatingOccupancyFraction=65 -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=0x7FFFFFFFFFFFFE -Dsun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE -XX:-OmitStackTraceInFastThrow -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -Djdk.tls.ephemeralDHKeySize=3072 -Djdk.tls.rejectClientInitiatedRenegotiation=true -Djava.io.tmpdir=${Bigdata_tmp_dir}

The average number of blocks stored in each DataNode instance in the cluster is: Number of HDFS blocks x 3/Number of DataNodes. If the average number of blocks changes, you need to change **-Xms2G -Xmx4G -XX:NewSize=128M -XX:MaxNewSize=256M** in the default value. The following table lists the reference values.

Table 12-199 DataNode JVM configuration

Average Number of Blocks in a DataNode Instance	Reference Value
2,000,000	-Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
5,000,000	-Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G

Xmx specifies memory which corresponds to the threshold of the number of DataNode blocks, and each GB memory supports a maximum of 500,000 DataNode blocks. Set the memory as required.

Viewing the HDFS Capacity Status

- NameNode information

  For versions earlier than MRS 3.*x*: Log in to the MRS console, and choose **Components** > **HDFS** > **NameNode (Active)**. Click **Overview** and check the number of file objects, files, directories, or blocks in the HDFS in **Summary**.

  For MRS 3.*x* or later: Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **NameNode(Active)**, and click **Overview** to view information like the number of file objects, files, directories, and blocks in HDFS in **Summary** area.

- DataNode information

  For versions earlier than MRS 3.*x*: Log in to the MRS console and choose **Components** > **HDFS** > **NameNode (Active)**. Click **DataNodes** and check the number of blocks of all DataNodes that report alarms.

  For MRS 3.*x* or later: Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **NameNode(Active)**, and click **DataNodes** to view the number of blocks on all DataNodes that report alarms.

- Alarm information

Check whether the alarms whose IDs are 14007, 14008, and 14009 are generated and change the alarm thresholds as required.

12.9.11 Configuring ulimit for HBase and HDFS

Symptom

When you open an HDFS file, an error occurs due to the limit on the number of file handles. Information similar to the following is displayed.

```
IOException (Too many open files)
```

Procedure

You can contact the administrator to add file handles for each user. This is a configuration on the OS instead of HBase or HDFS. It is recommended that the administrator configure the number of file handles based on the service traffic of HBase and HDFS and the rights of each user. If a user performs a large number of operations frequently on the HDFS that has large service traffic, set the number of file handles of this user to a large value.

Step 1 Log in to the OSs of all nodes or clients in the cluster as user **root**, and go to the **/etc/security** directory.

Step 2 Run the following command to edit the **limits.conf** file:

vi limits.conf

Add the following information to the file.

```
hdfs -       nofile  32768
hbase -      nofile  32768
```

hdfs and **hbase** indicate the usernames of the OSs that are used during the services.

NOTE

- Only user **root** has the rights to edit the **limits.conf** file.
- If this modification does not take effect, check whether other nofile values exist in the **/etc/security/limits.d** directory. Such values may overwrite the values set in the **/etc/security/limits.conf** file.
- If a user needs to perform operations on HBase, set the number of file handles of this user to a value greater than **10000**. If a user needs to perform operations on HDFS, set the number of file handles of this user based on the service traffic. It is recommended that the value not be too small. If a user needs to perform operations on both HBase and HDFS, set the number of file handles of this user to a large value, such as **32768**.

Step 3 Run the following command to check the limit on the number of file handles of a user:

su - *user_name*

ulimit -n

The limit on the number of file handles of this user is displayed as follows.

```
8194
```

----End

12.9.12 Balancing DataNode Capacity

Scenario

NOTE

This section applies to MRS 3.*x* or later.

In the HDFS cluster, unbalanced disk usage among DataNodes may occur, for example, when new DataNodes are added to the cluster. Unbalanced disk usage may result in multiple problems. For example, MapReduce applications cannot make full use of local computing advantages, network bandwidth usage between data nodes cannot be optimal, or node disks cannot be used. Therefore, the system administrator needs to periodically check and maintain DataNode data balance.

HDFS provides a capacity balancing program Balancer. By running Balancer, you can balance the HDFS cluster and ensure that the difference between the disk usage of each DataNode and that of the HDFS cluster does not exceed the threshold. DataNode disk usage before and after balancing is shown in **Figure 12-18** and **Figure 12-19**, respectively.

Figure 12-18 DataNode disk usage before balancing

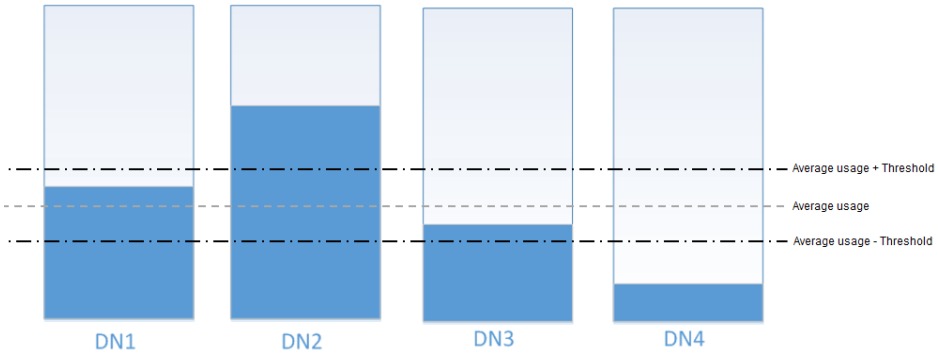
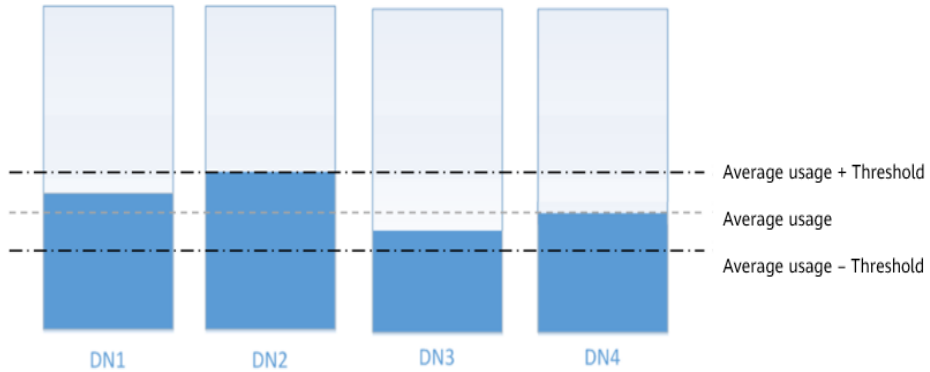


Figure 12-19 DataNode disk usage after balancing



The time of the balancing operation is affected by the following two factors:

1. Total amount of data to be migrated:

   The data volume of each DataNode must be greater than (Average usage - Threshold) x Average data volume and less than (Average usage + Threshold)

x Average data volume. If the actual data volume is less than the minimum value or greater than the maximum value, imbalance occurs. The system sets the largest deviation volume on all DataNodes as the total data volume to be migrated.

2. Balancer migration is performed in sequence in iteration mode. The amount of data to be migrated in each iteration does not exceed 10 GB, and the usage of each iteration is recalculated.

Therefore, for a cluster, you can estimate the time consumed by each iteration (by observing the time consumed by each iteration recorded in balancer logs) and divide the total data volume by 10 GB to estimate the task execution time.

The balancer can be started or stopped at any time.

Impact on the System

- The balance operation occupies network bandwidth resources of DataNodes. Perform the operation during maintenance based on service requirements.
- The balance operation may affect the running services if the bandwidth traffic (the default bandwidth control is 20 MB/s) is reset or the data volume is increased.

Prerequisites

The client has been installed.

Procedure

Step 1 Log in to the node where the client is installed as a client installation user. Run the following command to switch to the client installation directory, for example, **/opt/client**:

cd /opt/client

NOTE

If the cluster is in normal mode, run the **su - omm** command to switch to user **omm**.

Step 2 Run the following command to configure environment variables:

source bigdata_env

Step 3 If the cluster is in security mode, run the following command to authenticate the HDFS identity:

kinit hdfs

Step 4 Determine whether to adjust the bandwidth control.

- If yes, go to **Step 5**.
- If no, go to **Step 6**.

Step 5 Run the following command to change the maximum bandwidth of Balancer, and then go to **Step 6**.

hdfs dfsadmin -setBalancerBandwidth ***<bandwidth in bytes per second>***

<bandwidth in bytes per second> indicates the bandwidth control value, in bytes. For example, to set the bandwidth control to 20 MB/s (the corresponding value is 20971520), run the following command:

hdfs dfsadmin -setBalancerBandwidth 20971520

NOTE

- The default bandwidth control is 20 MB/s. This value is applicable to the scenario where the current cluster uses the 10GE network and services are being executed. If the service idle time window is insufficient for balance maintenance, you can increase the value of this parameter to shorten the balance time, for example, to 209715200 (200 MB/s).
- The value of this parameter depends on the networking. If the cluster load is high, you can change the value to 209715200 (200 MB/s). If the cluster is idle, you can change the value to 1073741824 (1 GB/s).
- If the bandwidth of the DataNodes cannot reach the specified maximum bandwidth, modify the HDFS parameter **dfs.datanode.balance.max.concurrent.moves** on FusionInsight Manager, and change the number of threads for balancing on each DataNode to **32** and restart the HDFS service.

Step 6 Run the following command to start the balance task:

bash /opt/client/HDFS/hadoop/sbin/start-balancer.sh -threshold *<threshold of balancer>*

-threshold specifies the deviation value of the DataNode disk usage, which is used for determining whether the HDFS data is balanced. When the difference between the disk usage of each DataNode and the average disk usage of the entire HDFS cluster is less than this threshold, the system considers that the HDFS cluster has been balanced and ends the balance task.

For example, to set deviation rate to 5%, run the following command:

bash /opt/client/HDFS/hadoop/sbin/start-balancer.sh -threshold 5

NOTE

- The preceding command executes the task in the background. You can query related logs in the **hadoop-root-balancer-***host name***.out log** file in the **/opt/client/HDFS/hadoop/logs** directory of the host.
- To stop the balance task, run the following command:

  bash /opt/client/HDFS/hadoop/sbin/stop-balancer.sh
- If only data on some nodes needs to be balanced, you can add the **-include** parameter in the script to specify the nodes to be migrated. You can run commands to view the usage of different parameters.
- **/opt/client** is the client installation directory. If the directory is inconsistent, replace it.
- If the command fails to be executed and the error information **Failed to APPEND_FILE /system/balancer.id** is displayed in the log, run the following command to forcibly delete **/system/balancer.id** and run the **start-balancer.sh** script again:

  hdfs dfs -rm -f /system/balancer.id

Step 7 If the following information is displayed, the balancing is complete and the system automatically exits the task:

```
Apr 01, 2016 01:01:01 PM  Balancing took 23.3333 minutes
```

After you run the script in **Step 6**, the **hadoop-root-balancer-***Host name***.out log** file is generated in the client installation directory **/opt/client/HDFS/hadoop/logs**. You can view the following information in the log:

- Time Stamp
- Bytes Already Moved
- Bytes Left To Move
- Bytes Being Moved

----End

Related Tasks

Enable automatic execution of the balance task

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations**, select **All Configurations**, search for the following parameters, and change the parameter values.

- **dfs.balancer.auto.enable** indicates whether to enable automatic balance task execution. The default value **false** indicates that automatic balance task execution is disabled. The value **true** indicates that automatic execution is enabled.
- **dfs.balancer.auto.cron.expression** indicates the task execution time. The default value **0 1 * * 6** indicates that the task is executed at 01:00 every Saturday. This parameter is valid only when the automatic execution is enabled.

  Table 12-200 describes the expression for modifying this parameter. * indicates consecutive time segments.

  Table 12-200 Parameters in the execution expression

Column	Description
1	Minute. The value ranges from 0 to 59.
2	Hour. The value ranges from 0 to 23.
3	Date. The value ranges from 1 to 31.
4	Month. The value ranges from 1 to 12.
5	Week. The value ranges from 0 to 6. **0** indicates Sunday.

- **dfs.balancer.auto.stop.cron.expression** indicates the task ending time. The default value is empty, indicating that the running balance task is not automatically stopped. For example, **0 5 * * 6** indicates that the balance task is stopped at 05:00 every Saturday. This parameter is valid only when the automatic execution is enabled.

  Table 12-200 describes the expression for modifying this parameter. * indicates consecutive time segments.

Step 3 Running parameters of the balance task that is automatically executed are shown in Table 12-201.

Table 12-201 Running parameters of the automatic balancer

Parameter	Parameter description	Default Value
dfs.balancer.auto.threshold	Specifies the balancing threshold of the disk capacity percentage. This parameter is valid only when **dfs.balancer.auto.enable** is set to **true**.	10
dfs.balancer.auto.exclude.datanodes	Specifies the list of DataNodes on which automatic disk balancing is not required. This parameter is valid only when **dfs.balancer.auto.enable** is set to **true**.	The value is left blank by default.
dfs.balancer.auto.bandwidthPerSec	Specifies the maximum bandwidth (MB/s) of each DataNode for load balancing.	20
dfs.balancer.auto.maxIdleIterations	Specifies the maximum number of consecutive idle iterations of Balancer. An idle iteration is an iteration without moving blocks. When the number of consecutive idle iterations reaches the maximum number, the balance task ends. The value **-1** indicates infinity.	5
dfs.balancer.auto.maxDataNodesNum	Controls the number of DataNodes that perform automatic balance tasks. Assume that the value of this parameter is *N*. If *N* is greater than 0, data is balanced between *N* DataNodes with the highest percentage of remaining space and *N* DataNodes with the lowest percentage of remaining space. If *N* is 0, data is balanced among all DataNodes in the cluster.	5

Step 4 Click **Save** to make configurations take effect. You do not need to restart the HDFS service.

Go to the **/var/log/Bigdata/hdfs/nn/hadoop-omm-balancer-***Host name***.log** file to view the task execution logs saved in the active NameNode.

----End

12.9.13 Configuring Replica Replacement Policy for Heterogeneous Capacity Among DataNodes

Scenario

By default, NameNode randomly selects a DataNode to write files. If the disk capacity of some DataNodes in a cluster is inconsistent (the total disk capacity of some nodes is large and of some nodes is small), the nodes with small disk capacity will be fully written. To resolve this problem, change the default disk selection policy for data written to DataNode to the available space block policy. This policy increases the probability of writing data blocks to the node with large available disk space. This ensures that the node usage is balanced when disk capacity of DataNodes is inconsistent.

Impact on the System

The disk selection policy is changed to **org.apache.hadoop.hdfs.server.blockmanagement.AvailableSpaceBlockPlacementPolicy**. It is proven that the HDFS file write performance optimizes by 3% after the modification.

NOTE

The default replica storage policy of the NameNode is as follows:

1. First replica: stored on the node where the client resides.
2. Second replica: stored on DataNodes of the remote rack.
3. Third replica: stored on different nodes of the same rack for the node where the client resides.

If there are more replicas, randomly store them on other DataNodes.

The replica selection mechanism (**org.apache.hadoop.hdfs.server.blockmanagement.AvailableSpaceBlockPlacementPolicy**) is as follows:

1. First replica: stored on the DataNode where the client resides (the same as the default storage policy).
2. Second replica:
   - When selecting a storage node, select two data nodes that meet the requirements.
   - Compare the disk usages of the two DataNodes. If the difference is smaller than 5%, store the replicas to the first node.
   - If the difference exceeds 5%, there is a 60% probability (specified by **dfs.namenode.available-space-block-placement-policy.balanced-space-preference-fraction** and default value is **0.6**) that the replica is written to the node whose disk space usage is low.
3. As for the storage of the third replica and subsequent replicas, refer to that of the second replica.

Prerequisites

The total disk capacity deviation of DataNodes in the cluster cannot exceed 100%.

Procedure

Step 1 Go to the **All Configurations** page of HDFS by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 Modify the disk selection policy parameters when HDFS writes data. Search for the **dfs.block.replicator.classname** parameter and change its value to **org.apache.hadoop.hdfs.server.blockmanagement.AvailableSpaceBlockPlacementPolicy**.

Step 3 Save the modified configuration. Restart the expired service or instance for the configuration to take effect.

----End

12.9.14 Configuring the Number of Files in a Single HDFS Directory

Scenario

Generally, multiple services are deployed in a cluster, and the storage of most services depends on the HDFS file system. Different components such as Spark and Yarn or clients are constantly writing files to the same HDFS directory when the cluster is running. However, the number of files in a single directory in HDFS is limited. Users must plan to prevent excessive files in a single directory and task failure.

You can set the number of files in a single directory using the **dfs.namenode.fs-limits.max-directory-items** parameter in HDFS.

Procedure

Step 1 Go to the **All Configurations** page of HDFS by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 Search for the configuration item **dfs.namenode.fs-limits.max-directory-items**.

Table 12-202 Parameter description

Parameter	Description	Default Value
dfs.namenode.fs-limits.max-directory-items	Maximum number of items in a directory Value range: 1 to 6,400,000	1048576

Step 3 Set the maximum number of files that can be stored in a single HDFS directory. Save the modified configuration. Restart the expired service or instance for the configuration to take effect.

NOTE

Plan data storage in advance based on time and service type categories to prevent excessive files in a single directory. You are advised to use the default value, which is about 1 million pieces of data in a single directory.

----End

12.9.15 Configuring the Recycle Bin Mechanism

Scenario

On HDFS, deleted files are moved to the recycle bin (trash can) so that the data deleted by mistake can be restored.

You can set the time threshold for storing files in the recycle bin. Once the file storage duration exceeds the threshold, it is permanently deleted from the recycle bin. If the recycle bin is cleared, all files in the recycle bin are permanently deleted.

Configuration Description

If a file is deleted from HDFS, the file is saved in the trash space rather than cleared immediately. After the aging time is due, the deleted file becomes an aging file and will be cleared based on the system mechanism or manually cleared by users.

Parameter portal:

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-203 Parameter description

Parameter	Description	Default Value
fs.trash.interval	Trash collection time, in minutes. If data in the trash station exceeds the time, the data will be deleted. Value range: 1440 to 259200	1440
fs.trash.checkpoint.interval	Interval between trash checkpoints, in minutes. The value must be less than or equal to the value of **fs.trash.interval**. The checkpoint program creates a checkpoint every time it runs and removes the checkpoint created **fs.trash.interval** minutes ago. For example, the system checks whether aging files exist every 10 minutes and deletes aging files if any. Files that are not aging are stored in the checkpoint list waiting for the next check. If this parameter is set to 0, the system does not check aging files and all aging files are saved in the system. Value range: 0 to *fs.trash.interval* **NOTE** It is not recommended to set this parameter to 0 because aging files will use up the disk space of the cluster.	60

12.9.16 Setting Permissions on Files and Directories

Scenario

HDFS allows users to modify the default permissions of files and directories. The default mask provided by the HDFS for creating file and directory permissions is **022**. If you have special requirements for the default permissions, you can set configuration items to change the default permissions.

Configuration Description

Parameter portal:

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-204 Parameter description

Parameter	Description	Default Value
fs.permissions.umask-mode	This **umask** value (user mask) is used when the user creates files and directories in the HDFS on the clients. This parameter is similar to the file permission mask on Linux. The parameter value can be in octal or in symbolic, for example, **022** (octal, the same as **u=rwx,g=r-x,o=r-x** in symbolic), or **u=rwx,g=rwx,o=** (symbolic, the same as **007** in octal). **NOTE** The octal mask is opposite to the actual permission value. You are advised to use the symbol notation to make the description clearer.	022

12.9.17 Setting the Maximum Lifetime and Renewal Interval of a Token

Scenario

In security mode, users can flexibly set the maximum token lifetime and token renewal interval in HDFS based on cluster requirements.

Configuration Description

Navigation path for setting parameters:

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-205 Parameter description

Parameter	Description	Default Value
dfs.namenode.delegation.token.max-lifetime	This parameter is a server parameter. It specifies the maximum lifetime of a token. Unit: milliseconds. Value range: 10,000 to 10,000,000,000,000	604,800,000
dfs.namenode.delegation.token.renew-interval	This parameter is a server parameter. It specifies the maximum lifetime to renew a token. Unit: milliseconds. Value range: 10,000 to 10,000,000,000,000	86,400,000

12.9.18 Configuring the Damaged Disk Volume

Scenario

In the open source version, if multiple data storage volumes are configured for a DataNode, the DataNode stops providing services by default if one of the volumes is damaged. You can change the value of **dfs.datanode.failed.volumes.tolerated** to specify the number of damaged disk volumes that are allowed. If the number of damaged volumes does not exceed the threshold, DataNode continues to provide services.

Configuration Description

Navigation path for setting parameters:

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-206 Parameter description

Parameter	Description	Default Value
dfs.datanode.failed.volumes.tolerated	Specifies the number of damaged volumes that are allowed before the DataNode stops providing services. By default, there must be at least one valid volume. The value **-1** indicates that the minimum value of a valid volume is **1**. The value greater than or equal to **0** indicates the number of damaged volumes that are allowed.	Versions earlier than MRS 3.x: 0 MRS 3.*x* or later: -1

12.9.19 Configuring Encrypted Channels

Scenario

Encrypted channel is an encryption protocol of remote procedure call (RPC) in HDFS. When a user invokes RPC, the user's login name will be transmitted to RPC through RPC head. Then RPC uses Simple Authentication and Security Layer (SASL) to determine an authorization protocol (Kerberos and DIGEST-MD5) to complete RPC authorization. When users deploy security clusters, they need to use encrypted channels and configure the following parameters. For details about the secure Hadoop RPC, visit **https://hadoop.apache.org/docs/r3.1.1/hadoop-project-dist/hadoop-common/SecureMode.html#Data_Encryption_on_RPC**.

Configuration Description

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-207 Parameter description

Parameter	Description	Default Value
hadoop.rpc.protection	**NOTICE** • The setting takes effect only after the service is restarted. Rolling restart is not supported. • After the setting, you need to download the client configuration again. Otherwise, the HDFS cannot provide the read and write services. Whether the RPC channels of each module in Hadoop are encrypted. The channels include: • RPC channels for clients to access HDFS • RPC channels between modules in HDFS, for example, RPC channels between DataNode and NameNode • RPC channels for clients to access Yarn • RPC channels between NodeManager and ResourceManager • RPC channels for Spark to access Yarn and HDFS • RPC channels for MapReduce to access Yarn and HDFS • RPC channels for HBase to access HDFS **NOTE** You can set this parameter on the HDFS component configuration page. The parameter setting takes effect globally, that is, the setting of whether the RPC channel is encrypted takes effect on all modules in Hadoop. There are three encryption modes. • **authentication**: This is the default value in normal mode. In this mode, data is directly transmitted without encryption after being authenticated. This mode ensures performance but has security risks. • **integrity**: Data is transmitted without encryption or authentication. To ensure data security, exercise caution when using this mode. • **privacy**: This is the default value in security mode, indicating that data is transmitted after authentication and encryption. This mode reduces the performance.	• Security mode: privacy • Normal mode: authentication

12.9.20 Reducing the Probability of Abnormal Client Application Operation When the Network Is Not Stable

Scenario

Clients probably encounter running errors when the network is not stable. Users can adjust the following parameter values to improve the running efficiency.

Configuration Description

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-208 Parameter description

Parameter	Description	Default Value
ha.health-monitor.rpc-timeout.ms	Timeout interval during the NameNode health check performed by ZKFC. Increasing this value can prevent dual active NameNodes and reduce the probability of application running exceptions on clients. Unit: millisecond. Value range: 30,000 to 3,600,000	180,000
ipc.client.connect.max.retries.on.timeouts	Number of retry times when the socket connection between a server and a client times out. Value range: 1 to 256	45
ipc.client.connect.timeout	Timeout interval of the socket connection between a client and a server. Increasing the value of this parameter increases the timeout interval for setting up a connection. Unit: millisecond. Value range: 1 to 3,600,000	20,000

12.9.21 Configuring the NameNode Blacklist

Scenario

NOTE

This section applies to MRS 3.*x* or later.

In the existing default DFSclient failover proxy provider, if a NameNode in a process is faulty, all HDFS client instances in the same process attempt to connect to the NameNode again. As a result, the application waits for a long time and timeout occurs.

When clients in the same JVM process connect to the NameNode that cannot be accessed, the system is overloaded. The NameNode blacklist is equipped with the MRS cluster to avoid this problem.

In the new Blacklisting DFSClient failover provider, the faulty NameNode is recorded in a list. The DFSClient then uses the information to prevent the client from connecting to such NameNodes again. This function is called NameNode blacklisting.

For example, there is a cluster with the following configurations:

namenode: nn1, nn2

dfs.client.failover.connection.retries: 20

Processes in a single JVM: 10 clients

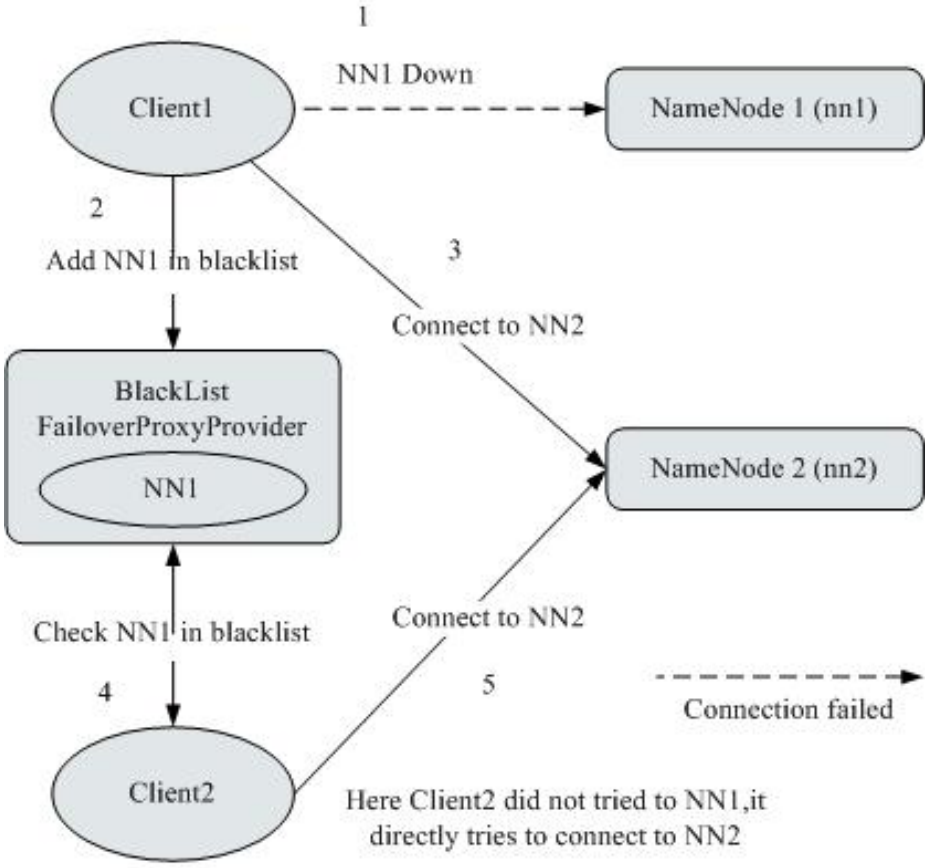
In the preceding cluster, if the active **nn1** cannot be accessed, client1 will retry the connection for 20 times. Then, a failover occurs, and client1 will connect to **nn2**. In the same way, other clients also connect to **nn2** when the failover occurs after retrying the connection to **nn1** for 20 times. Such process prolongs the fault recovery of NameNode.

In this case, the NameNode blacklisting adds **nn1** to the blacklist when client1 attempts to connect to the active **nn1** which is already faulty. Therefore, other clients will avoid trying to connect to **nn1** but choose **nn2** directly.

NOTE

If, at any time, all NameNodes are added to the blacklist, the content in the blacklist will be cleared, and the client attempts to connect to the NameNodes based on the initial NameNode list. If any fault occurs again, the NameNode is still added to the blacklist.

Figure 12-20 NameNode blacklisting working principle



Configuration Description

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-209 NameNode blacklisting parameters

Parameter	Description	Default Value
dfs.client.failover.proxy.provider.*[nameserviceID]*	Client Failover proxy provider class which creates the NameNode proxy using the authenticated protocol. Set this parameter to **org.apache.hadoop.hdfs.server.namenode.ha.BlackListingFailoverProxyProvider**. You can configure the observer NameNode to process read requests.	org.apache.hadoop.hdfs.server.namenode.ha.AdaptiveFailoverProxyProvider

12.9.22 Optimizing HDFS NameNode RPC QoS

Scenarios

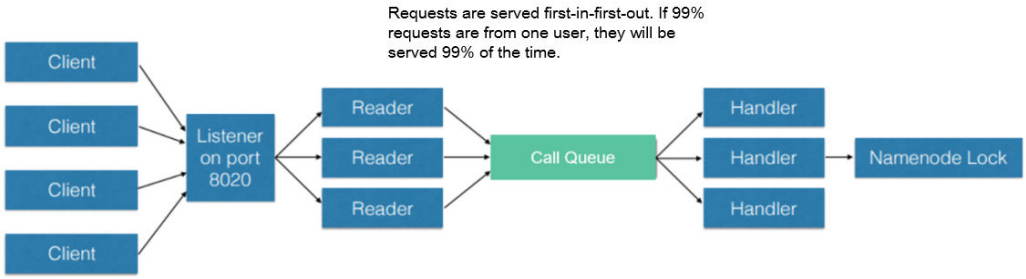
NOTE

This section applies to MRS 3.*x* or later.

Several finished Hadoop clusters are faulty because the NameNode is overloaded and unresponsive.

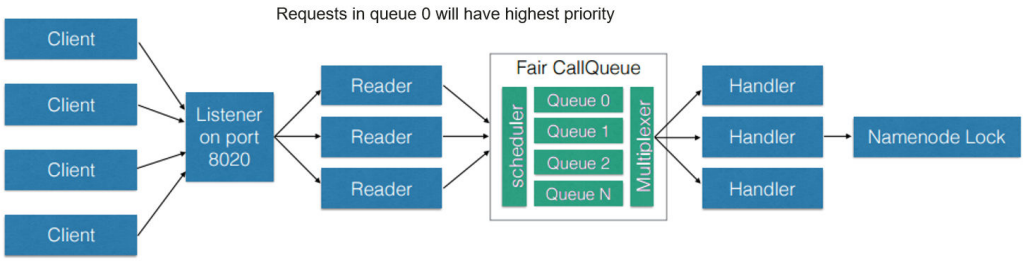
Such problem is caused by the initial design of Hadoop: In Hadoop, the NameNode functions as an independent part and in its namespace coordinates various HDFS operations, including obtaining the data block location, listing directories, and creating files. The NameNode receives HDFS operations, regards them as RPC calls, and places them in the FIFO call queue for read threads to process. Requests in FIFO call queue are served first-in first-out. However, users who perform more I/O operations are served more time than those performing fewer I/O operations. In this case, the FIFO is unfair and causes the delay.

Figure 12-21 NameNode request processing based on the FIFO call queue



The unfair problem and delaying mentioned before can be improved by replacing the FIFO queue with a new type of queue called FairCallQueue. In this way, FAIR queues assign incoming RPC calls to multiple queues based on the scale of the caller's call. The scheduling module tracks the latest calls and assigns a higher priority to users with a smaller number of calls.

Figure 12-22 NameNode request processing based on FAIRCallQueue



Configuration Description

- FairCallQueue ensures quality of service (QoS) by internally adjusting the order in which RPCs are invoked.

  This queue consists of the following parts:

a. DecayRpcScheduler: used to provide priority values from 0 to N (the value 0 indicates the highest priority).
b. Multi-level queues (located in the FairCallQueue): used to ensure that queues are invoked in order of priority.
c. Multi-channel converters (provided with Weighted Round Robin Multiplexer): used to provide logic control for queue selection.

After the FairCallQueue is configured, the control module determines the sub-queue to which the received invoking is allocated. The current scheduling module is DecayRpcScheduler, which only continuously tracks the priority numbers of various calls and periodically reduces these numbers.

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-210 FairCallQueue parameters

Parameter	Description	Default Value
ipc.*<port>*.callqueue.impl	Specifies the queue implementation class. You need to run the **org.apache.hadoop.ipc.FairCallQueue** command to enable the QoS feature.	java.util.concurrent.LinkedBlockingQueue

- RPC BackOff

  Backoff is one of the FairCallQueue functions. It requires the client to retry operations (such as creating, deleting, and opening a file) after a period of time. When the backoff occurs, the RCP server throws RetriableException. The FairCallQueue performs backoff in either of the following cases:

  – The queue is full, that is, there are many client calls in the queue.
  – The queue response time is longer than the threshold time (specified by the **ipc.<port>.decay-scheduler.backoff.responsetime.thresholds** parameter).

Table 12-211 RPC Backoff configuration

Parameter	Description	Default Value
ipc.*<port>*.backoff.enable	Specifies whether to enable the backoff. When the current application contains a large number of user callings, the RPC request is blocked if the connection limit of the operating system is not reached. Alternatively, when the RPC or NameNode is heavily loaded, some explicit exceptions can be thrown back to the client based on certain policies. The client can understand these exceptions and perform exponential rollback, which is another implementation of the RetryInvocationHandler class.	false
ipc.*<port>*.decay-scheduler.backoff.responsetime.enable	Indicate whether to enable the backoff based on the average queue response time.	false
ipc.*<port>*.decay-scheduler.backoff.responsetime.thresholds	Configure the response time threshold for each queue. The response time threshold must match the number of priorities (the value of **ipc.<port> .faircallqueue.priority-levels**). Unit: millisecond	10000,20000,30000,40000

NOTE

- **<port>** indicates the RPC port configured on the NameNode.
- The backoff function based on the response time takes effect only when **ipc.<port> .backoff.enable** is set to **true**.

12.9.23 Optimizing HDFS DataNode RPC QoS

Scenario

When the speed at which the client writes data to the HDFS is greater than the disk bandwidth of the DataNode, the disk bandwidth is fully occupied. As a result, the DataNode does not respond. The client can back off only by canceling or restoring the channel, which results in write failures and unnecessary channel recovery operations.

NOTE

This section applies to MRS 3.*x* or later.

Configuration

The new configuration parameter **dfs.pipeline.ecn** is introduced. When this configuration is enabled, the DataNode sends a signal from the write channel when the write channel is overloaded. The client may perform backoff based on the blocking signal to prevent the system from being overloaded. This configuration parameter is introduced to make the channel more stable and reduce unnecessary cancellation or recovery operations. After receiving the signal, the client backs off for a period of time (5,000 ms), and then adjusts the backoff time based on the related filter (the maximum backoff time is 50,000 ms).

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-212 DN ECN configuration

Parameter	Description	Default Value
dfs.pipeline.ecn	After configuration, the DataNode can send blocking notifications to the client.	false

12.9.24 Configuring Reserved Percentage of Disk Usage on DataNodes

Scenario

When the Yarn local directory and DataNode directory are on the same disk, the disk with larger capacity can run more tasks. Therefore, more intermediate data is stored in the Yarn local directory.

Currently, you can set **dfs.datanode.du.reserved** to configure the absolute value of the reserved disk space on DataNodes. A small value cannot meet the requirements of a disk with large capacity. However, configuring a large value for a disk with same capacity wastes a lot of disk space.

To avoid this problem, a new parameter **dfs.datanode.du.reserved.percentage** is introduced to configure the reserved percentage of the disk space.

NOTE

- If **dfs.datanode.du.reserved.percentage** and **dfs.datanode.du.reserved** are configured at the same time, the larger value of the reserved disk space calculated using the two parameters is used as the reserved space of the data nodes.
- You are advised to set **dfs.datanode.du.reserved** or **dfs.datanode.du.reserved.percentage** based on the actual disk space.

Configuration Description

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-213 Parameter description

Parameter	Description	Default Value
dfs.datanode.du.reserved.percentage	Indicates the percentage of the reserved disk space on DataNodes. The DataNode permanently reserves the disk space calculated using this percentage. The value is an integer ranging from 0 to 100.	10

12.9.25 Configuring HDFS NodeLabel

Scenario

You need to configure the nodes for storing HDFS file data blocks based on data features. You can configure a label expression to an HDFS directory or file and assign one or more labels to a DataNode so that file data blocks can be stored on specified DataNodes.

If the label-based data block placement policy is used for selecting DataNodes to store the specified files, the DataNode range is specified based on the label expression. Then proper nodes are selected from the specified range.

NOTE

This section applies to MRS 3.*x* or later.

After cross-AZ HA is enabled for a single cluster, the HDFS NodeLabel function cannot be configured.

- Scenario 1: DataNodes partitioning scenario

  Scenario description:

  When different application data is required to run on different nodes for separate management, label expressions can be used to achieve separation of different services, storing specified services on corresponding nodes.

By configuring the NodeLabel feature, you can perform the following operations:

- Store data in **/HBase** to DN1, DN2, DN3, and DN4.
- Store data in **/Spark** to DN5, DN6, DN7, and DN8.

Figure 12-23 DataNode partitioning scenario

NOTE

- Run the **hdfs nodelabel -setLabelExpression -expression 'LabelA[fallback=NONE]' -path /Hbase** command to set an expression for the **Hbase** directory. As shown in **Figure 12-23**, the data block replicas of files in the **/Hbase** directory are placed on the nodes labeled with the **LabelA**, that is, DN1, DN2, DN3, and DN4. Similarly, run the **hdfs nodelabel -setLabelExpression -expression 'LabelB[fallback=NONE]' -path /Spark** command to set an expression for the Spark directory. Data block replicas of files in the **/Spark** directory can be placed only on nodes labeled with **LabelB**, that is, DN5, DN6, DN7, and DN8.
- For details about how to set labels for a data node, see **Configuration Description**.
- If multiple racks are available in one cluster, it is recommended that DataNodes of these racks should be available under each label, to ensure reliability of data block placement.

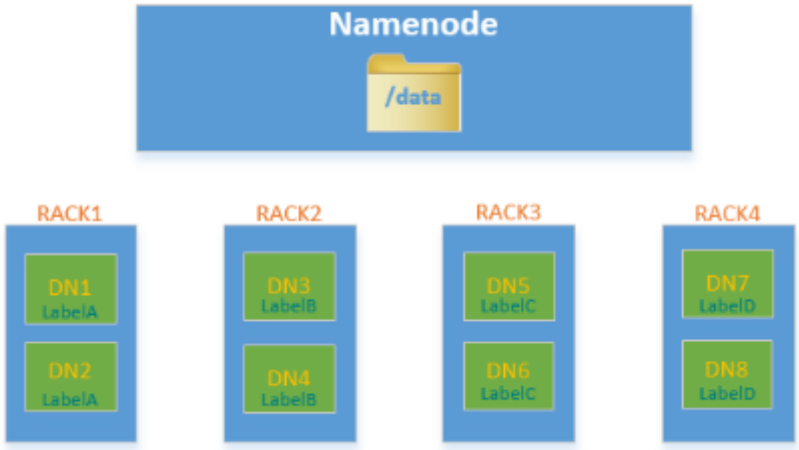
- Scenario 2: Specifying replica location when there are multiple racks

  Scenario description:

  In a heterogeneous cluster, customers need to allocate certain nodes with high availability to store important commercial data. Label expressions can be used to specify replica location so that the replica can be placed on a high reliable node.

  Data blocks in the **/data** directory have three replicas by default. In this case, at least one replica is stored on a node of RACK1 or RACK2 (nodes of RACK1 and RACK2 are high reliable), and the other two are stored separately on the nodes of RACK3 and RACK4.

Figure 12-24 Scenario example



NOTE

Run the **hdfs nodelabel -setLabelExpression -expression 'LabelA|| LabelB[fallback=NONE],LabelC,LabelD' -path /data** command to set an expression for the **/data** directory.

When data is to be written to the **/data** directory, at least one data block replica is stored on a node labeled with the LabelA or LabelB, and the other two data block replicas are stored separately on the nodes labeled with the LabelC and LabelD.

Configuration Description

- DataNode label configuration

  Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

  Table 12-214 Parameter description

Parameter	Description	Default Value
dfs.block.replicator.classname	Used to configure the DataNode policy of HDFS. To enable the NodeLabel function, set this parameter to **org.apache.hadoop.hdfs.server.blockmanagement.BlockPlacementPolicyWithNodeLabel**.	org.apache.hadoop.hdfs.server.blockmanagement.AvailableSpaceBlockPlacementPolicy

Parameter	Description	Default Value
host2tags	Used to configure a mapping between a DataNode host and a label. The host name can be configured with an IP address extension expression (for example, **192.168.1.[1-128]** or **192.168.[2-3].[1-128]**) or a regular expression (for example, **/datanode-[123]/** or **/datanode-\d{2}/**) starting and ending with a slash (/). The label configuration name cannot contain the following characters: = / \ **Note**: The IP address must be a service IP address.	-

NOTE

- The **host2tags** configuration item is described as follows:

  Assume there are 20 DataNodes which range from dn-1 to dn-20 in a cluster and the IP addresses of clusters range from 10.1.120.1 to 10.1.120.20. The value of **host2tags** can be represented in either of the following methods:

  Regular expression of the host name

  /dn-\d/ = label-1 indicates that the labels corresponding to dn-1 to dn-9 are label-1, that is, dn-1 = label-1, dn-2 = label-1, ..., dn-9 = label-1.

  /dn-((1[0-9]$)|(20$))/ = label-2 indicates that the labels corresponding to dn-10 to dn-20 are label-2, that is, dn-10 = label-2, dn-11 = label-2, ...dn-20 = label-2.

  IP address range expression

  10.1.120.[1-9] = label-1 indicates that the labels corresponding to 10.1.120.1 to 10.1.120.9 are label-1, that is, 10.1.120.1 = label-1, 10.1.120.2 = label-1, ..., and 10.1.120.9 = label-1.

  10.1.120.[10-20] = label-2 indicates that the labels corresponding to 10.1.120.10 to 10.1.120.20 are label-2, that is, 10.1.120.10 = label-2, 10.1.120.11 = label-2, ..., and 10.1.120.20 = label-2.

- Label-based data block placement policies are applicable to capacity expansion and reduction scenarios.

  A newly added DataNode will be assigned a label if the IP address of the DataNode is within the IP address range in the **host2tags** configuration item or the host name of the DataNode matches the host name regular expression in the **host2tags** configuration item.

  For example, the value of **host2tags** is **10.1.120.[1-9] = label-1**, but the current cluster has only three DataNodes: 10.1.120.1 to 10.1.120.3. If DataNode 10.1.120.4 is added for capacity expansion, the DataNode is labeled as label-1. If the 10.1.120.3 DataNode is deleted or out of the service, no data block will be allocated to the node.

- Set label expressions for directories or files.
  - On the HDFS parameter configuration page, configure **path2expression** to configure the mapping between HDFS directories and labels. If the configured HDFS directory does not exist, the configuration can succeed. When a directory with the same name as the HDFS directory is created manually, the configured label mapping relationship will be inherited by the directory within 30 minutes. After a labeled directory is deleted, a

new directory with the same name as the deleted one will inherit its mapping within 30 minutes.

- For details about configuring items using commands, see the **hdfs nodelabel -setLabelExpression** command.
- To set label expressions using the Java API, invoke the **setLabelExpression(String src, String labelExpression)** method using the instantiated object NodeLabelFileSystem. *src* indicates a directory or file path on HDFS, and **labelExpression** indicates the label expression.

- After the NodeLabel is enabled, you can run the **hdfs nodelabel -listNodeLabels** command to view the label information of each DataNode.

Block Replica Location Selection

Nodelabel supports different placement policies for replicas. The expression **label-1,label-2,label-3** indicates that three replicas are respectively placed in DataNodes containing label-1, label-2, and label-3. Different replica policies are separated by commas (,).

If you want to place two replicas in DataNode with label-1, set the expression as follows: **label-1[replica=2],label-2,label-3**. In this case, if the default number of replicas is 3, two nodes with label-1 and one node with label-2 are selected. If the default number of replicas is 4, two nodes with label-1, one node with label-2, and one node with label-3 are selected. Note that the number of replicas is the same as that of each replica policy from left to right. However, the number of replicas sometimes exceeds the expressions. If the default number of replicas is 5, the extra replica is placed on the last node, that is, the node labeled with label-3.

When the ACLs function is enabled and the user does not have the permission to access the labels used in the expression, the DataNode with the label is not selected for the replica.

Deletion of Redundant Block Replicas

If the number of block replicas exceeds the value of **dfs.replication** (number of file replicas specified by the user), HDFS will delete redundant block replicas to ensure cluster resource usage.

The deletion rules are as follows:

- Preferentially delete replicas that do not meet any expression.

  For example: The default number of file replicas is **3**.

  The label expression of **/test** is **LA[replica=1],LB[replica=1],LC[replica=1]**.

  The file replicas of **/test** are distributed on four nodes (D1 to D4), corresponding to labels (LA to LD).

  ```
  D1:LA
  D2:LB
  D3:LC
  D4:LD
  ```

  Then, block replicas on node D4 will be deleted.

- If all replicas meet the expressions, delete the redundant replicas which are beyond the number specified by the expression.

  For example: The default number of file replicas is **3**.

The label expression of **/test** is **LA[replica=1],LB[replica=1],LC[replica=1]**.

The file replicas of **/test** are distributed on the following four nodes, corresponding to the following labels.

```
D1:LA
D2:LA
D3:LB
D4:LC
```

Then, block replicas on node D1 or D2 will be deleted.

- If a file owner or group of a file owner cannot access a label, preferentially delete the replica from the DataNode mapped to the label.

Example of label-based block placement policy

Assume that there are six DataNodes, namely, dn-1, dn-2, dn-3, dn-4, dn-5, and dn-6 in a cluster and the corresponding IP address range is 10.1.120.[1-6]. Six directories must be configured with label expressions. The default number of block replicas is **3**.

- The following provides three expressions of the DataNode label in **host2labels** file. The three expressions have the same function.
  - Regular expression of the host name

    ```
    /dn-[1456]/ = label-1,label-2
    /dn-[26]/ = label-1,label-3
    /dn-[3456]/ = label-1,label-4
    /dn-5/ = label-5
    ```

  - IP address range expression

    ```
    10.1.120.[1-6] = label-1
    10.1.120.1 = label-2
    10.1.120.2 = label-3
    10.1.120.[3-6] = label-4
    10.1.120.[4-6] = label-2
    10.1.120.5 = label-5
    10.1.120.6 = label-3
    ```

  - Common host name expression

    ```
    /dn-1/ = label-1, label-2
    /dn-2/ = label-1, label-3
    /dn-3/ = label-1, label-4
    /dn-4/ = label-1, label-2, label-4
    /dn-5/ = label-1, label-2, label-4, label-5
    /dn-6/ = label-1, label-2, label-3, label-4
    ```

- The label expressions of the directories are set as follows:

  ```
  /dir1 = label-1
  /dir2 = label-1 && label-3
  /dir3 = label-2 || label-4[replica=2]
  /dir4 = (label-2 || label-3) && label-4
  /dir5 = !label-1
  /sdir2.txt = label-1 && label-3[replica=3,fallback=NONE]
  /dir6 = label-4[replica=2],label-2
  ```

  NOTE

  For details about the label expression configuration, see the **hdfs nodelabel -setLabelExpression** command.

  The file data block storage locations are as follows:

  - Data blocks of files in the **/dir1** directory can be stored on any of the following nodes: dn-1, dn-2, dn-3, dn-4, dn-5, and dn-6.
  - Data blocks of files in the **/dir2** directory can be stored on the dn-2 and dn-6 nodes. The default number of block replicas is **3**. The expression

matches only two DataNodes. The third replica will be stored on one of the remaining nodes in the cluster.

- Data blocks of files in the **/dir3** directory can be stored on any three of the following nodes: dn-1, dn-3, dn-4, dn-5, and dn-6.
- Data blocks of files in the **/dir4** directory can be stored on the dn-4, dn-5, and dn-6 nodes.
- Data blocks of files in the **/dir5** directory do not match any DataNode and will be stored on any three nodes in the cluster, which is the same as the default block selection policy.
- For the data blocks of the **/sdir2.txt** file, two replicas are stored on the dn-2 and dn-6 nodes. The left one is not stored in the node because **fallback=NONE** is enabled.
- Data blocks of the files in the **/dir6** directory are stored on the two nodes with label-4 selected from dn-3, dn-4, dn-5, and dn-6 and another node with label-2. If the specified number of file replicas in the **/dir6** directory is more than 3, the extra replicas will be stored on a node with label-2.

Restrictions

In configuration files, **key** and **value** are separated by equation signs (=), colons (:), and whitespace. Therefore, the host name of the **key** cannot contain these characters because these characters may be considered as separators.

12.9.26 Configuring HDFS Mover

Scenario

Mover is a new data migration tool whose working mode is similar to that of the HDFS Balancer. Mover can redistribute data in the cluster based on the configured data storage policy.

Use Mover to periodically check whether the specified HDFS file or directory in the HDFS file system meets the preset storage policy. If not, migrate data to make them meet the policy.

NOTE

This section applies to MRS 3.*x* or later.

Configuration Description

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-215 Parameter description

Parameter	Description	Default Value
dfs.mover.auto.enable	Specifies whether to enable the data replica migration function. This function supports multiple modes. The default value is **false**, indicating that this function is disabled.	false
dfs.mover.auto.cron.expression	Specifies the CRON expression for HDFS automatic data migration, and is used to control the start time of data migration. This parameter is valid only when **dfs.mover.auto.enable** is set to **true**. The default value is **0 * * * ***, indicating that the task is executed on the hour. For details about CRON expression, see **Table 12-216**.	0 * * * *
dfs.mover.auto.hdfsfiles_or_dirs	Specifies HDFS file and directory lists that implement automatic replica migration in specified clusters. Multiple values are separated by space. This parameter is valid only when **dfs.mover.auto.enable** is set to **true**.	-

Table 12-216 CRON expressions

Column	Description
1	Minute. The value ranges from 0 to 59.
2	Hour. The value ranges from 0 to 23.
3	Date. The value ranges from 1 to 31.
4	Month. The value ranges from 1 to 12.
5	Week. The value ranges from 0 to 6. **0** indicates Sunday.

Use Restrictions

Run the command on the HDFS client to enable the mover function. The command format is as follows:

hdfs mover -p *<Full path or directory path of an HDFS file >*

NOTE

Users running this command on the client must have the **supergroup** permission. You can use the system user **hdfs** of the HDFS service. Alternatively, you can create a user with the **supergroup** permission in the cluster and then run the command.

12.9.27 Using HDFS AZ Mover

Scenario

AZ Mover is a copy migration tool used to move copies to meet the new AZ policies set on the directory. It can be used to migrate copies from one AZ policy to another. AZ Mover instructs NameNode to move copies based on a new AZ policy. If the NameNode refuses to delete the old copies, the new policy may not be met. For example, the copies are marked as outdated.

Restrictions

- Changing the policy name to **LOCAL_AZ** is the same as that to **ONE_AZ** because the client location cannot be determined when the uploaded file is written.
- Mover cannot determine the AZ status. As a result, the copy may be moved to the abnormal AZ and depends on NameNode for further processing.
- Mover depends on whether the number of DataNodes in each AZ meets the minimum requirement. If the AZ Mover is executed in an AZ with a small number of DataNodes, the result may be different from the expected result.
- Mover only meets the AZ-level policies and does not guarantee to meet the basic block placement policy (BPP).
- Mover does not support the change of replication factors. If the number of copies in the new AZ is different from that in the old AZ, an exception occurs.

Procedure

Step 1 Run the following command to go to the client installation directory.

cd /opt/client

Step 2 Run the following command to configure environment variables:

source bigdata_env

Step 3 If the cluster is in security mode, the user must have the read permission on the source directory or file and the write permission on the destination directory, and run the following command to authenticate the user: In normal mode, skip user authentication.

kinit *Component service user*

Step 4 Create a directory and set an AZ policy.

Run the following command to create a directory.

hdfs dfs -mkdir <*path*>

Run the following command to set the AZ policy (**azexpression** indicates the AZ policy):

hdfs dfsadmin -setAZExpression *<path> <azexpression>*

Run the following command to view the AZ policy:

hdfs dfsadmin -getAZExpression *<path>*

Step 5 Upload files to the directory.

hdfs dfs -put <*localfile*> <*hdfs-path*>

Step 6 Delete the old policy from the directory and set a new policy.

Run the following command to clear the old policy:

hdfs dfsadmin -clearAZExpression <*path*>

Run the following command to configure a new policy:

hdfs dfsadmin -setAZExpression <*path*> <*azexpression*>

Step 7 Run the **azmover** command to make the copy distribution meet the new AZ policy.

hdfs azmover -p /targetDirecotry

----End

12.9.28 Configuring HDFS DiskBalancer

Scenario

DiskBalancer is an online disk balancer that balances disk data on running DataNodes based on various indicators. It works in the similar way of the HDFS Balancer. The difference is that HDFS Balancer balances data between DataNodes, while HDFS DiskBalancer balances data among disks on a single DataNode.

Data among disks may be unevenly distributed if a large number of files have been deleted from a cluster running for a long time, or disk capacity expansion is performed on a node in the cluster. Uneven data distribution may deteriorate the concurrent read/write performance of the HDFS, or cause service failure due to inappropriate HDFS write policies. In this case, the data density among disks on a node needs to be balanced to prevent heterogeneous small disks from becoming the performance bottleneck of the node.

NOTE

This section applies to MRS 3.*x* or later.

Configuration Description

Go to the **All Configurations** page of HDFS and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-217 Parameter description

Parameter	Description	Default Value
dfs.disk.balancer.auto.enabled	Indicates whether to enable the HDFS DiskBalancer function. The default value is **false**, indicating that this function is disabled.	false

Parameter	Description	Default Value
dfs.disk.balancer.auto.cron.expression	CRON expression of the HDFS disk balancing operation, which is used to control the start time of the balancing operation. This parameter is valid only when **dfs.disk.balancer.auto.enabled** is set to **true**. The default value is **0 1 * * 6**, indicating that tasks are executed at 01:00 every Saturday. For details about cron expression, see **Table 12-218**. The default value indicates that the DiskBalancer check is executed at 01:00 every Saturday.	0 1 * * 6
dfs.disk.balancer.max.disk.throughputInMBperSec	Specifies the maximum disk bandwidth that can be used for disk data balancing. The unit is MB/s, and the default value is **10**. Set this parameter based on the actual disk conditions of the cluster.	10
dfs.disk.balancer.max.disk.errors	Specifies the maximum number of errors that are allowed in a specified movement process. If the value exceeds this threshold, the movement fails.	5
dfs.disk.balancer.block.tolerance.percent	Specifies the difference threshold between the data storage capacity and perfect status of each disk during data balancing among disks. For example, the ideal data storage capacity of each disk is 1 TB, and this parameter is set to **10**. When the data storage capacity of the target disk reaches 900 GB, the storage status of the disk is considered as perfect. Value range: 1 to 100.	10
dfs.disk.balancer.plan.threshold.percent	Specifies the data density difference that is allowed between two disks during disk data balancing. If the absolute value of the data density difference between any two disks exceeds the threshold, data balancing is required. Value range: 1 to 100.	10
dfs.disk.balancer.top.nodes.number	Specifies the top *N* nodes whose disk data needs to be balanced in the cluster.	5

To use this function, set **dfs.disk.balancer.auto.enabled** to **true** and configure a proper CRON expression. Set other parameters based on the cluster status.

Table 12-218 CRON expressions

Column	Description
1	Minute. The value ranges from 0 to 59.
2	Hour. The value ranges from 0 to 23.
3	Date. The value ranges from 1 to 31.
4	Month. The value ranges from 1 to 12.
5	Week. The value ranges from 0 to 6. **0** indicates Sunday.

Use Restrictions

1. Data can only be moved between disks of the same type. For example, data can only be moved between SSDs or between DISKs.
2. Enabling this function occupies disk I/O resources and network bandwidth resources of involved nodes. Enable this function in off-peak hours.
3. The DataNodes specified by the **dfs.disk.balancer.top.nodes.number** parameter are frequently calculated. Therefore, set the parameter to a small value.
4. Commands for using the DiskBalancer function on the HDFS client are as follows:

   Table 12-219 DiskBalancer commands

Syntax	Description
hdfs diskbalancer -report -top <N>	Set *N* to an integer greater than 0. This command can be used to query the top *N* nodes that require disk data balancing in the cluster.
hdfs diskbalancer -plan <Hostname|IP Address>	This command can be used to generate a JSON file based on the DataNode. The file contains information about the source disk, target disk, and blocks to be moved. In addition, this command can be used to specify other parameters such as the network bandwidth.
hdfs diskbalancer -query <Hostname:$dfs.datanode.ipc.port>	The default port number of the cluster is 9867. This command is used to query the running status of the DiskBalancer task on the current node.

Syntax	Description
hdfs diskbalancer -execute <planfile>	In this command, **planfile** indicates the JSON file generated in the second command. Use the absolute path.
hdfs diskbalancer -cancel <planfile>	This command is used to cancel the running planfile. Use the absolute path.

NOTE

- Users running this command on the client must have the **supergroup** permission. You can use the system user **hdfs** of the HDFS service. Alternatively, you can create a user with the **supergroup** permission in the cluster and then run the command.
- Only formats and usage of commands are provided in **Table 12-219**. For more parameters to be configured for each command, run the **hdfs diskbalancer -help <command>** command to view detailed information.
- When you troubleshoot performance problems during the cluster O&M, check whether the HDFS disk balancing occurs in the event information of the cluster. If yes, check whether DiskBalancer is enabled in the cluster.
- After the automatic DiskBalancer function is enabled, the ongoing task stops only after the current data balancing is complete. The task cannot be canceled during the balancing.
- You can manually specify certain nodes for data balancing on the client.

12.9.29 Configuring the Observer NameNode to Process Read Requests

Scenario

In an HDFS cluster configured with HA, the active NameNode processes all client requests, and the standby NameNode reserves the latest metadata and block location information. However, in this architecture, the active NameNode is the bottleneck of client request processing. This bottleneck is more obvious in clusters with a large number of requests.

To address this issue, a new NameNode is introduced: an observer NameNode. Similar to the standby NameNode, the observer NameNode also reserves the latest metadata information and block location information. In addition, the observer NameNode can process read requests from clients in the same way as the active NameNode. In typical HDFS clusters with many read requests, the observer NameNode can be used to process read requests, reducing the active NameNode load and improving the cluster capability of processing requests.

NOTE

This section applies to MRS 3.*x* or later.

Impact on the System

- The active NameNode load can be reduced and the capability of HDFS cluster processing requests can be improved, which is especially obvious for large clusters.
- The client application configuration needs to be updated.

Prerequisites

- The HDFS cluster has been installed, the active and standby NameNodes are running properly, and the HDFS service is normal.
- The **${BIGDATA_DATA_HOME}/namenode** partition has been created on the node where the observer NameNode is to be installed.

Procedure

The following steps describe how to configure the observer NameNode of a hacluster and enable it to process read requests. If there are multiple pairs of NameServices in the cluster and they are all in use, perform the following steps to configure the observer NameNode for each pair.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **NameService Management**.

Step 3 Click **Add** next to **hacluster**.

Step 4 On the **Add NameNode** page, set **NameNode type** to **Observer** and click **Next**.

Step 5 On the **Assign Role** page, select the planned host, add the observer NameNode, and click **Next**.

NOTE

A maximum of five observer NameNodes can be added to each pair of NameServices.

Step 6 On the configuration page, configure the storage directory and port number of the NameNode as planned and click **Next**.

Step 7 Confirm the information, click **Submit**, and wait until the installation of the observer NameNode is complete.

Step 8 Restart the upper-layer components that depend on HDFS, update the client application configuration, and restart the client application.

----End

12.9.30 Performing Concurrent Operations on HDFS Files

Scenario

Performing this operation can concurrently modify file and directory permissions and access control tools in a cluster.

NOTE

This section applies to MRS 3.*x* or later.

Impact on the System

Performing concurrent file modification operations in a cluster has adverse impacts on the cluster performance. Therefore, you are advised to do so when the cluster is idle.

Prerequisites

- The HDFS client or clients including HDFS has been installed. For example, the installation directory is **/opt/client**.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user needs to change the password upon the first login. (This operation is not required in normal mode.)

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, the user executing the DistCp command must belong to the **supergroup** group and run the following command to perform user authentication. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Increase the JVM size of the client to prevent out of memory (OOM). (32 GB is recommended for 100 million files.)

NOTE

The HDFS client exits abnormally and the error message "java.lang.OutOfMemoryError" is displayed after the HDFS client command is executed.

This problem occurs because the memory required for running the HDFS client exceeds the preset upper limit (128 MB by default). You can change the memory upper limit of the client by modifying **CLIENT_GC_OPTS** in *<Client installation path>***/HDFS/component_env**. For example, if you want to set the upper limit to 1 GB, run the following command:

CLIENT_GC_OPTS="-Xmx1G"

After the modification, run the following command to make the modification take effect:

source <*Client installation path*>/**/bigdata_env**

Step 6 Run the concurrent commands shown in the following table.

Command	Description	Function
hdfs quickcmds [-t threadsNumber] [-p principal] [-k keytab] -setrep <rep> <path> ...	**threadsNumber** indicates the number of concurrent threads. The default value is the number of vCPUs of the local host. **principal** indicates the Kerberos user. **keytab** indicates the Keytab file. **rep** indicates the number of replicas. **path** indicates the HDFS directory.	Used to concurrently set the number of copies of all files in a directory.
hdfs quickcmds [-t threadsNumber] [-p principal] [-k keytab] -chown [owner][:[group]] <path> ...	**threadsNumber** indicates the number of concurrent threads. The default value is the number of vCPUs of the local host. **principal** indicates the Kerberos user. **keytab** indicates the Keytab file. **owner** indicates the owner. **group** indicates the group to which the user belongs. **path** indicates the HDFS directory.	Used to concurrently set the owner group of all files in the directory.
hdfs quickcmds [-t threadsNumber] [-p principal] [-k keytab] -chmod <mode> <path> ...	**threadsNumber** indicates the number of concurrent threads. The default value is the number of vCPUs of the local host. **principal** indicates the Kerberos user. **keytab** indicates the Keytab file. **mode** indicates the permission (for example, 754). **path** indicates the HDFS directory.	Used to concurrently set permissions for all files in a directory.
hdfs quickcmds [-t threadsNumber] [-p principal] [-k keytab] -setfacl [{-b|-k} {-m|-x <acl_spec>} <path> ...]|[--set <acl_spec> <path> ...]	**threadsNumber** indicates the number of concurrent threads. The default value is the number of vCPUs of the local host. **principal** indicates the Kerberos user. **keytab** indicates the Keytab file. **acl_spec** indicates the ACL list separated by commas (,). **path** indicates the HDFS directory.	Used to concurrently set ACL information for all files in a directory.

----End

12.9.31 Introduction to HDFS Logs

Log Description

Log path: The default path of HDFS logs is **/var/log/Bigdata/hdfs/***Role name*.

- NameNode: **/var/log/Bigdata/hdfs/nn** (run logs) and **/var/log/Bigdata/audit/hdfs/nn** (audit logs)
- DataNode: **/var/log/Bigdata/hdfs/dn** (run logs) and **/var/log/Bigdata/audit/hdfs/dn** (audit logs)
- ZKFC: **/var/log/Bigdata/hdfs/zkfc** (run logs) and **/var/log/Bigdata/audit/hdfs/zkfc** (audit logs)
- JournalNode: **/var/log/Bigdata/hdfs/jn** (run logs) and **/var/log/Bigdata/audit/hdfs/jn** (audit logs)
- Router: **/var/log/Bigdata/hdfs/router** (run logs) and **/var/log/Bigdata/audit/hdfs/router** (audit logs)
- HttpFS: **/var/log/Bigdata/hdfs/httpfs** (run logs) and **/var/log/Bigdata/audit/hdfs/httpfs** (audit logs)

Log archive rule: The automatic HDFS log compression function is enabled. By default, when the size of logs exceeds 100 MB, logs are automatically compressed into a log file named in the following format: *<Original log file name>-<yyyy-mm-dd_hh-mm-ss.[ID]>***.log.zip**. A maximum of 100 latest compressed files are reserved. The number of compressed files can be configured on Manager.

Table 12-220 HDFS log list

Type	Name	Description
Run log	hadoop-<SSH_USER>-<process_name>-<hostname>.log	HDFS system log, which records most of the logs generated when the HDFS system is running.
	hadoop-<SSH_USER>-<process_name>-<hostname>.out	Log that records the HDFS running environment information.
	hadoop.log	Log that records the operation of the Hadoop client.

Type	Name	Description
	hdfs-period-check.log	Log that records scripts that are executed periodically, including automatic balancing, data migration, and JournalNode data synchronization detection.
	<process_name>-<SSH_USER>-<DATE>-<PID>-gc.log	Garbage collection log file
	postinstallDetail.log	Work log before the HDFS service startup and after the installation.
	hdfs-service-check.log	Log that records whether the HDFS service starts successfully.
	hdfs-set-storage-policy.log	Log that records the HDFS data storage policies.
	cleanupDetail.log	Log that records the cleanup logs about the uninstallation of the HDFS service.
	prestartDetail.log	Log that records cluster operations before the HDFS service startup.
	hdfs-recover-fsimage.log	Recovery log of the NameNode metadata.
	datanode-disk-check.log	Log that records the disk status check during the cluster installation and use.
	hdfs-availability-check.log	Log that check whether the HDFS service is available.
	hdfs-backup-fsimage.log	Backup log of the NameNode metadata.
	startDetail.log	Detailed log that records the HDFS service startup.

Type	Name	Description
	hdfs-blockplacement.log	Log that records the placement policy of HDFS blocks.
	upgradeDetail.log	Upgrade logs.
	hdfs-clean-acls-java.log	Log that records the clearing of deleted roles' ACL information by HDFS.
	hdfs-haCheck.log	Run log that checks whether the NameNode in active or standby state has obtained scripts.
	<process_name>-jvmpause.log	Log that records JVM pauses during process running.
	hadoop-<SSH_USER>-balancer-<hostname>.log	Run log of HDFS automatic balancing.
	hadoop-<SSH_USER>-balancer-<hostname>.out	Log that records information of the environment where HDFS executes automatic balancing.
	hdfs-switch-namenode.log	Run log that records the HDFS active/standby switchover.
	hdfs-router-admin.log	Run log of the mount table management operation
Tomcat logs	hadoop-omm-host1.out, httpfs-catalina.<DATE>.log, httpfs-host-manager.<DATE>.log, httpfs-localhost.<DATE>.log, httpfs-manager.<DATE>.log, localhost_access_web_log.log	Tomcat run log
Audit log	hdfs-audit-<process_name>.log ranger-plugin-audit.log	Audit log that records the HDFS operations (such as creating, deleting, modifying and querying files).
	SecurityAuth.audit	HDFS security audit log.

Log Level

Table 12-221 lists the log levels supported by HDFS. The log levels include FATAL, ERROR, WARN, INFO, and DEBUG. Logs of which the levels are higher than or equal to the set level will be printed by programs. The higher the log level is set, the fewer the logs are recorded.

Table 12-221 Log levels

Level	Description
FATAL	Indicates the critical error information about system running.
ERROR	Indicates the error information about system running.
WARN	Indicates that the current event processing exists exceptions.
INFO	Indicates that the system and events are running properly.
DEBUG	Indicates the system and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of HDFS by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the left menu bar, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

NOTE

The configurations take effect immediately without restarting the service.

----End

Log Formats

The following table lists the HDFS log formats.

Table 12-222 Log formats

Type	Format	Example
Run log	*<yyyy-MM-dd HH:mm:ss,SSS>*|*<Log level>*|*<Name of the thread that generates the log>*|*<Message in the log>*|*<Location where the log event occurs>*	2015-01-26 18:43:42,840 | INFO | IPC Server handler 40 on 8020 | Rolling edit logs | org.apache.hadoop.hdfs.server.namenode.FSEditLog.rollEditLog(FSEditLog.java:1096)
Audit log	*<yyyy-MM-dd HH:mm:ss,SSS>*|*<Log level>*|*<Name of the thread that generates the log>*|*<Message in the log>*|*<Location where the log event occurs>*	2015-01-26 18:44:42,607 | INFO | IPC Server handler 32 on 8020 | allowed=true ugi=hbase (auth:SIMPLE) ip=/10.177.112.145 cmd=getfileinfo src=/hbase/WALs/hghoulaslx410,16020,1421743096083/hghoulaslx410%2C16020%2C1421743096083.1422268722795 dst=null perm=null | org.apache.hadoop.hdfs.server.namenode.FSNamesystem$DefaultAuditLogger.logAuditMessage(FSNamesystem.java:7950)

12.9.32 HDFS Performance Tuning

12.9.32.1 Improving Write Performance

Scenario

Improve the HDFS write performance by modifying the HDFS attributes.

NOTE

This section applies to MRS 3.*x* or later.

Procedure

Navigation path for setting parameters:

On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations** and select **All Configurations**. Enter a parameter name in the search box.

Table 12-223 Parameters for improving HDFS write performance

Parameter	Description	Default Value
dfs.datanode.drop.cache.behind.reads	Specifies whether to enable a DataNode to automatically clear all data in the cache after the data in the cache is transferred to the client. If it is set to **true**, the cached data is discarded. The parameter needs to be configured on the DataNode. You are advised to set it to **true** if data is repeatedly read only a few times, so that the cache can be used by other operations. You are advised to set it to **false** if data is repeatedly read many times to enhance the reading speed.	false
dfs.client-write-packet-size	Specifies the size of the client write packet. When the HDFS client writes data to the DataNode, the data will be accumulated until a packet is generated. Then, the packet is transmitted over the network. This parameter specifies the size (unit: byte) of the data packet to be transmitted, which can be specified by each job. In the 10-Gigabit network, you can increase the value of this parameter to enhance the transmission throughput.	262144

12.9.32.2 Improving Read Performance Using Client Metadata Cache

Scenario

Improve the HDFS read performance by using the client to cache the metadata for block locations.

NOTE

This function is recommended only for reading files that are not modified frequently. Because the data modification done on the server side by some other client is invisible to the cache client, which may cause the metadata obtained from the cache to be outdated.

This section applies to MRS 3.*x* or later.

Procedure

Navigation path for setting parameters:

On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations**, select **All Configurations**, and enter the parameter name in the search box.

Table 12-224 Parameter configuration

Parameter	Description	Default Value
dfs.client.metadata.cache.enabled	Enables or disables the client to cache the metadata for block locations. Set this parameter to **true** and use it along with the **dfs.client.metadata.cache.pattern** parameter to enable the cache.	false
dfs.client.metadata.cache.pattern	Indicates the regular expression pattern of the path of the file to be cached. Only the metadata for block locations of these files is cached until the metadata expires. This parameter is valid only when **dfs.client.metadata.cache.enabled** is set to **true**. Example: **/test.*** indicates that all files whose paths start with **/test** are read. **NOTE** • To ensure consistency, configure a specific mode to cache only files that are not frequently modified by other clients. • The regular expression pattern verifies only the path of the URI, but not the schema and authority in the case of the Fully Qualified path.	-
dfs.client.metadata.cache.expiry.sec	Indicates the duration for caching metadata. The cache entry becomes invalid after its caching time exceeds this duration. Even metadata that is frequently used during the caching process can become invalid. Time suffixes **s**/**m**/**h** can be used to indicate second, minute, and hour, respectively. **NOTE** If this parameter is set to **0s**, the cache function is disabled.	60s
dfs.client.metadata.cache.max.entries	Indicates the maximum number of non-expired data items that can be cached at a time.	65536

NOTE

Call *DFSClient#clearLocatedBlockCache()* to completely clear the client cache before it expires.

The sample usage is as follows:

```
FileSystem fs = FileSystem.get(conf);
DistributedFileSystem dfs = (DistributedFileSystem) fs;
DFSClient dfsClient = dfs.getClient();
dfsClient.clearLocatedBlockCache();
```

12.9.32.3 Improving the Connection Between the Client and NameNode Using Current Active Cache

Scenario

When HDFS is deployed in high availability (HA) mode with multiple NameNode instances, the HDFS client needs to connect to each NameNode in sequence to determine which is the active NameNode and use it for client operations.

Once the active NameNode is identified, its details can be cached and shared to all clients running on the client host. In this way, each new client first tries to load the details of the active Name Node from the cache and save the RPC call to the standby NameNode, which can help a lot in abnormal scenarios, for example, when the standby NameNode cannot be connected for a long time.

When a fault occurs and the other NameNode is switched to the active state, the cached details are updated to the information about the current active NameNode.

NOTE

This section applies to MRS 3.*x* or later.

Procedure

Navigation path for setting parameters:

On FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Configurations**, select **All Configurations**, and enter the parameter name in the search box.

Table 12-225 Configuration parameters

Parameter	Description	Default Value
dfs.client.failover.proxy.provider.[nameservice ID]	Client Failover proxy provider class which creates the NameNode proxy using the authenticated protocol. If this parameter is set to **org.apache.hadoop.hdfs.server.namenode.ha.BlackListingFailoverProxyProvider**, you can use the NameNode blacklist feature on the HDFS client. If this parameter is set to **org.apache.hadoop.hdfs.server.namenode.ha.ObserverReadProxyProvider**, you can configure the observer NameNode to process read requests.	org.apache.hadoop.hdfs.server.namenode.ha.AdaptiveFailoverProxyProvider
dfs.client.failover.activeinfo.share.flag	Specifies whether to enable the cache function and share the detailed information about the current active NameNode with other clients. Set it to **true** to enable the cache function.	false

Parameter	Description	Default Value
dfs.client.failover.activeinfo.share.path	Specifies the local directory for storing the shared files created by all clients in the host. If a cache area is to be shared by different users, the directory must have required permissions (for example, creating, reading, and writing cache files in the specified directory).	/tmp
dfs.client.failover.activeinfo.share.io.timeout.sec	(Optional) Used to control timeout. The cache file is locked when it is being read or written, and if the file cannot be locked within the specified time, the attempt to read or update the caches will be abandoned. The unit is second.	5

NOTE

The cache files created by the HDFS client are reused by other clients, and thus these files will not be deleted from the local system. If this function is disabled, you may need to manually clear the data.

12.9.33 FAQ

12.9.33.1 NameNode Startup Is Slow

Question

The NameNode startup is slow when it is restarted immediately after a large number of files (for example, 1 million files) are deleted.

Answer

It takes time for the DataNode to delete the corresponding blocks after files are deleted. When the NameNode is restarted immediately, it checks the block information reported by all DataNodes. If a deleted block is found, the NameNode generates the corresponding INFO log information, as shown below:

```
2015-06-10 19:25:50,215 | INFO  | IPC Server handler 36 on 25000 | BLOCK* processReport: blk_1075861877_2121067 on node 10.91.8.218:9866 size 10249 does not belong to any file | org.apache.hadoop.hdfs.server.blockmanagement.BlockManager.processReport(BlockManager.java:1854)
```

A log is generated for each deleted block. A file may contain one or more blocks. Therefore, after startup, the NameNode spends a large amount of time printing logs when a large number of files are deleted. As a result, the NameNode startup becomes slow.

To address this issue, the following operations can be performed to speed up the startup:

1. After a large number of files are deleted, wait until the DataNode deletes the corresponding blocks and then restart the NameNode.

You can run the ***hdfs dfsadmin -report*** command to check the disk space and check whether the files have been deleted.

2. If a large number of the preceding logs are generated, you can change the NameNode log level to **ERROR** so that the NameNode stops printing such logs.

   After the NameNode is restarted, change the log level back to **INFO**. You do not need to restart the service after changing the log level.

12.9.33.2 DataNode Is Normal but Cannot Report Data Blocks

Question

The DataNode is normal, but cannot report data blocks. As a result, the existing data blocks cannot be used.

Answer

This error may occur when the number of data blocks in a data directory exceeds four times the upper limit (4 x 1 MB). And the DataNode generates the following error logs:

```
2015-11-05 10:26:32,936 | ERROR | DataNode:[[[DISK]file:/srv/BigData/hadoop/data1/dn/]] heartbeating to
vm-210/10.91.8.210:8020 | Exception in BPOfferService for Block pool
BP-805114975-10.91.8.210-1446519981645
(Datanode Uuid bcada350-0231-413b-bac0-8c65e906c1bb) service to vm-210/10.91.8.210:8020 |
BPServiceActor.java:824
java.lang.IllegalStateException:com.google.protobuf.InvalidProtocolBufferException:Protocol message was
too large.May
be malicious.Use CodedInputStream.setSizeLimit() to increase the size limit. at
org.apache.hadoop.hdfs.protocol.BlockListAsLongs$BufferDecoder$1.next(BlockListAsLongs.java:369)
at org.apache.hadoop.hdfs.protocol.BlockListAsLongs$BufferDecoder$1.next(BlockListAsLongs.java:347) at
org.apache.hadoop.hdfs.
protocol.BlockListAsLongs$BufferDecoder.getBlockListAsLongs(BlockListAsLongs.java:325) at
org.apache.hadoop.hdfs.protocolPB.DatanodeProtocolClientSideTranslatorPB.
blockReport(DatanodeProtocolClientSideTranslatorPB.java:190) at
org.apache.hadoop.hdfs.server.datanode.BPServiceActor.blockReport(BPServiceActor.java:473)
at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.offerService(BPServiceActor.java:685) at
org.apache.hadoop.hdfs.server.datanode.BPServiceActor.run(BPServiceActor.java:822)
at java.lang.Thread.run(Thread.java:745) Caused
by:com.google.protobuf.InvalidProtocolBufferException:Protocol message was too large.May be
malicious.Use CodedInputStream.setSizeLimit()
to increase the size limit. at
com.google.protobuf.InvalidProtocolBufferException.sizeLimitExceeded(InvalidProtocolBufferException.java:1
10) at com.google.protobuf.CodedInputStream.refillBuffer(CodedInputStream.java:755)
at com.google.protobuf.CodedInputStream.readRawByte(CodedInputStream.java:769) at
com.google.protobuf.CodedInputStream.readRawVarint64(CodedInputStream.java:462) at
com.google.protobuf.
CodedInputStream.readSInt64(CodedInputStream.java:363) at
org.apache.hadoop.hdfs.protocol.BlockListAsLongs$BufferDecoder$1.next(BlockListAsLongs.java:363)
```

The number of data blocks in the data directory is displayed as **Metric**. You can monitor its value through **http://<datanode-ip>:<http-port>/jmx**. If the value is greater than four times the upper limit (4 x 1 MB), you are advised to configure multiple drives and restart HDFS.

Recovery procedure:

1. Configure multiple data directories on the DataNode.

   For example, configure multiple directories on the DataNode where only the **/data1/datadir** directory is configured:

```
<property> <name>dfs.datanode.data.dir</name> <value>/data1/datadir</value> </property>
```

Configure as follows:

```
<property> <name>dfs.datanode.data.dir</name> <value>/data1/datadir/,/data2/datadir,/data3/
datadir</value> </property>
```

NOTE

You are advised to configure multiple data directories on multiple disks. Otherwise, performance may be affected.

2. Restart the HDFS.
3. Perform the following operation to move the data to the new data directory:

   mv */data1/datadir/current/finalized/subdir1 /data2/datadir/current/finalized/subdir1*

4. Restart the HDFS.

12.9.33.3 HDFS WebUI Cannot Properly Update Information About Damaged Data

Question

1. When errors occur in the **dfs.datanode.data.dir** directory of DataNode due to the permission or disk damage, HDFS WebUI does not display information about damaged data.
2. After errors are restored, HDFS WebUI does not timely remove related information about damaged data.

Answer

1. DataNode checks whether the disk is normal only when errors occur in file operations. Therefore, only when a data damage is detected and the error is reported to NameNode, NameNode displays information about the damaged data on HDFS WebUI.
2. After errors are fixed, you need to restart DataNode. During restarting DataNode, all data states are checked and damaged data information is uploaded to NameNode. Therefore, after errors are fixed, damaged data information is not displayed on the HDFS WebUI only by restarting DataNode.

12.9.33.4 Why Does the Distcp Command Fail in the Secure Cluster, Causing an Exception?

Question

Why distcp command fails in the secure cluster with the following error displayed?

Client side exception

```
Invalid arguments: Unexpected end of file from server
```

Server side exception

```
javax.net.ssl.SSLException: Unrecognized SSL message, plaintext connection?
```

Answer

The preceding error may occur if **webhdfs://** is used in the distcp command. The reason is that the big data cluster uses the HTTPS mechanism, that is, **dfs.http.policy** is set to **HTTPS_ONLY** in **core-site.xml** file. To avoid the error, replace **webhdfs://** with **swebhdfs://** in the file.

For example:

./hadoop distcp swebhdfs://IP:PORT/testfile hdfs://IP:PORT/testfile1

12.9.33.5 Why Does DataNode Fail to Start When the Number of Disks Specified by dfs.datanode.data.dir Equals dfs.datanode.failed.volumes.tolerated?

Question

If the number of disks specified by **dfs.datanode.data.dir** is equal to the value of **dfs.datanode.failed.volumes.tolerated**, DataNode startup will fail.

Answer

By default, the failure of a single disk will cause the HDFS DataNode process to shut down, which results in the NameNode scheduling additional replicas for each block that is present on the DataNode. This causes needless replications of blocks that reside on disks that have not failed.

To prevent this, you can configure DataNodes to tolerate the failure of dfs.data.dir directories; use the **dfs.datanode.failed.volumes.tolerated** parameter in **hdfs-site.xml.** For example, if the value for this parameter is 3, the DataNode will only shut down after four or more data directories have failed. This value is respected on DataNode startup.

When we are configuring tolerate volumes which should be always less than the configured volumes or else we can keep this as -1 which is equal to n-1 (where n is number of disks) then DataNode will not be shut down.

12.9.33.6 Why Does an Error Occur During DataNode Capacity Calculation When Multiple data.dir Are Configured in a Partition?

Question

DataNode capacity count incorrect if several data.dir configured in one disk partition.

Answer

Currently calculation will be done based on the disk like ***df*** command in linux. Ideally user should not configure multiple directories for same disk which will be huge impact on performance where all data will go to one disk.

Hence it is always better to configure like below:

For example:

if the machine is having disks like following:

```
host-4:~ # df -h
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/sda1       352G  11G    324G  4%    /
udev            190G  252K  190G  1%    /dev
tmpfs           190G  72K    190G  1%    /dev/shm
/dev/sdb1       2.7T  74G    2.5T  3%    /data1
/dev/sdc1       2.7T  75G    2.5T  3%    /data2
/dev/sdd1       2.7T  73G    2.5T  3%    /data
```

Suggested way of configuration:

```
<property>
<name>dfs.datanode.data.dir</name>
<value>/data1/datadir/,/data2/datadir,/data3/datadir</value>
</property>
```

Following is not recommended:

```
<property>
<name>dfs.datanode.data.dir</name>
<value>/data1/datadir1/,/data2/datadir1,/data3/datadir1,/data1/datadir2/data1/datadir3,/data2/datadir2,/
data2/datadir3,/data3/datadir2,/data3/datadir3</value>
</property>
```

12.9.33.7 Standby NameNode Fails to Be Restarted When the System Is Powered off During Metadata (Namespace) Storage

Question

When the standby NameNode is powered off during metadata (namespace) storage, it fails to be started and the following error information is displayed.

```
2015-12-04 11:49:12,121 | ERROR | main | Failed to load image from FS
ImageFile(file=/srv/BigData/namenode/current/fsimage_0000000000000096
080,
cpktTxId=0000000000000096080) | FSImage.java:685
java.io.IOException: Invalid MD5 file /srv/BigData/namenode/current/f
simage_0000000000000096080.md5:
the content "                                   does not match the expecte
d pattern.
at org.apache.hadoop.hdfs.util.MD5FileUtils.readStoredMd5(MD5FileUtil
s.java:92)
at org.apache.hadoop.hdfs.util.MD5FileUtils.readStoredMd5ForFile(MD5F
ileUtils.java:109)
at org.apache.hadoop.hdfs.server.namenode.FSImage.loadFSImage(FSImage
.java:975)
at org.apache.hadoop.hdfs.server.namenode.FSImage.loadFSImageFile(FSI
mage.java:744)
at org.apache.hadoop.hdfs.server.namenode.FSImage.loadFSImage(FSImage
.java:682)
at org.apache.hadoop.hdfs.server.namenode.FSImage.recoverTransitionRe
ad(FSImage.java:300)
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFSImage(FS
Namesystem.java:968)
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFromDisk(F
SNamesystem.java:675)
at org.apache.hadoop.hdfs.server.namenode.NameNode.loadNamesystem(Nam
eNode.java:625)
at org.apache.hadoop.hdfs.server.namenode.NameNode.initialize(NameNod
e.java:685)
at org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.ja
va:889)
at org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.ja
va:872)
at org.apache.hadoop.hdfs.server.namenode.NameNode.createNameNode(Nam
eNode.java:1580)
at org.apache.hadoop.hdfs.server.namenode.NameNode.main(NameNode.java
:1654)
```

Answer

When the standby NameNode is powered off during metadata (namespace) storage, it fails to be started and the MD5 file is damaged. Remove the damaged fsimage and start the standby NameNode to rectify the fault. After the rectification, the standby NameNode loads the previous fsimage and reproduces all edits.

Recovery procedure:

1. Run the following command to remove the damaged fsimage:

   rm -rf ${BIGDATA_DATA_HOME}/namenode/current/fsimage_0000000000000096

2. Start the standby NameNode.

12.9.33.8 Why Data in the Buffer Is Lost If a Power Outage Occurs During Storage of Small Files

Question

Why data in the buffer is lost if a power outage occurs during storage of small files?

Answer

Because of a power outage, the blocks in the buffer are not written to the disk immediately after the write operation is completed. To enable synchronization of blocks to the disk, set **dfs.datanode.synconclose** to **true** in the **hdfs-site.xml** file.

By default, **dfs.datanode.synconclose** is set to **false**. This improves the performance but can cause a buffer data loss in the case of a power outage, and therefore, it is recommended that **dfs.datanode.synconclose** be set to **true** even if this may affect the performance. You can determine whether to enable the synchronization function based on your actual situation.

12.9.33.9 Why Does Array Border-crossing Occur During FileInputFormat Split?

Question

When HDFS calls the FileInputFormat getSplit method, the ArrayIndexOutOfBoundsException: 0 appears in the following log:

```
java.lang.ArrayIndexOutOfBoundsException: 0
at org.apache.hadoop.mapred.FileInputFormat.identifyHosts(FileInputFormat.java:708)
at org.apache.hadoop.mapred.FileInputFormat.getSplitHostsAndCachedHosts(FileInputFormat.java:675)
at org.apache.hadoop.mapred.FileInputFormat.getSplits(FileInputFormat.java:359)
at org.apache.spark.rdd.HadoopRDD.getPartitions(HadoopRDD.scala:210)
at org.apache.spark.rdd.RDD$$anonfun$partitions$2.apply(RDD.scala:239)
at org.apache.spark.rdd.RDD$$anonfun$partitions$2.apply(RDD.scala:237)
at scala.Option.getOrElse(Option.scala:120)
at org.apache.spark.rdd.RDD.partitions(RDD.scala:237)
at org.apache.spark.rdd.MapPartitionsRDD.getPartitions(MapPartitionsRDD.scala:35)
```

Answer

The elements of each block correspondent frame are as below: /default/rack0/:,/default/rack0/datanodeip:port.

The problem is due to a block damage or loss, making the block correspondent machine ip and port become null. Use **hdfs fsck** to check the file blocks health state when this problem occurs, and remove damaged block or restore the missing block to re-computing the task.

12.9.33.10 Why Is the Storage Type of File Copies DISK When the Tiered Storage Policy Is LAZY_PERSIST?

Question

When the storage policy of the file is set to **LAZY_PERSIST**, the storage type of the first replica should be **RAM_DISK**, and the storage type of other replicas should be **DISK**.

But why is the storage type of all copies shown as **DISK** actually?

Answer

When a user writes into a file whose storage policy is **LAZY_PERSIST**, three replicas are written one by one. The first replica is preferentially written into the

DataNode where the client is located. The storage type of all replicas is **DISK** in the following scenarios:

- If the DataNode where the client is located does not have the RAM disk, the first replica is written into the disk of the DataNode where the client is located, and other replicas are written into the disks of other nodes.
- If the DataNode where the client is located has the RAM disk, and the value of **dfs.datanode.max.locked.memory** is not specified or smaller than the value of **dfs.blocksize**, the first replica is written into the disk of the DataNode where the client is located, and other replicas are written into the disks of other nodes.

12.9.33.11 The HDFS Client Is Unresponsive When the NameNode Is Overloaded for a Long Time

Question

When the NameNode node is overloaded (100% of the CPU is occupied), the NameNode is unresponsive. The HDFS clients that are connected to the overloaded NameNode fail to run properly. However, the HDFS clients that are newly connected to the NameNode will be switched to a backup NameNode and run properly.

Answer

The default configuration must be used (as described in **Table 12-226**) when the error preceding described occurs: the **keep alive** mechanism is enabled for the RPC connection between the HDFS client and the NameNode. The **keep alive** mechanism will keep the HDFS client waiting for the response from server and prevent the connection from being out timed, causing the unresponsiveness of the HDFS client.

Perform the following operations to the unresponsive HDFS client:

- Leave the HDFS client waiting. Once the CPU usage of the node where NameNode locates drops, the NameNode will obtain CPU resources and the HDFS client will receive a response.
- If you do not want to leave the HDFS client running, restart the application where the HDFS client locates to reconnect the HDFS client to another idle NameNode.

Procedure:

Configure the following parameters in the **core-site.xml** file on the client.

Table 12-226 Parameter description

Parameter	Description	Default Value
ipc.client.ping	If the **ipc.client.ping** parameter is configured to **true**, the HDFS client will wait for the response from the server and periodically send the **ping** message to avoid disconnection caused by **tcp timeout**. If the **ipc.client.ping** parameter is configured to **false**, the HDFS client will set the value of **ipc.ping.interval** as the timeout time. If no response is received within that time, timeout occurs. To avoid the unresponsiveness of HDFS when the NameNode is overloaded for a long time, you are advised to set the parameter to **false**.	true
ipc.ping.interval	If the value of **ipc.client.ping** is **true**, **ipc.ping.interval** indicates the interval between sending the ping messages. If the value of **ipc.client.ping** is **false**, **ipc.ping.interval** indicates the timeout time for connection. To avoid the unresponsiveness of HDFS when the NameNode is overloaded for a long time, you are advised to set the parameter to a large value, for example **900000** (unit ms) to avoid timeout when the server is busy.	60000

12.9.33.12 Can I Delete or Modify the Data Storage Directory in DataNode?

Question

- In DataNode, the storage directory of data blocks is specified by **dfs.datanode.data.dir.** Can I modify **dfs.datanode.data.dir** to modify the data storage directory?
- Can I modify files under the data storage directory?

Answer

During the system installation, you need to configure the **dfs.datanode.data.dir** parameter to specify one or more root directories.

- During the system installation, you need to configure the dfs.datanode.data.dir parameter to specify one or more root directories.
- Exercise caution when modifying dfs.datanode.data.dir. You can configure this parameter to add a new data root directory.
- Do not modify or delete data blocks in the storage directory. Otherwise, the data blocks will lose.

NOTE

Similarly, do not delete the storage directory, or modify or delete data blocks under the directory using the following parameters:

- dfs.namenode.edits.dir
- dfs.namenode.name.dir
- dfs.journalnode.edits.dir

12.9.33.13 Blocks Miss on the NameNode UI After the Successful Rollback

Question

Why are some blocks missing on the NameNode UI after the rollback is successful?

Answer

This problem occurs because blocks with new IDs or genstamps may exist on the DataNode. The block files in the DataNode may have different generation flags and lengths from those in the rollback images of the NameNode. Therefore, the NameNode rejects these blocks in the DataNode and marks the files as damaged.

Scenarios:

1. Before an upgrade:

   Client A writes some data to file X. (Assume A bytes are written.)

2. During an upgrade:

   Client A still writes data to file X. (The data in the file is A + B bytes.)

3. After an upgrade:

   Client A completes the file writing. The final data is A + B bytes.

4. Rollback started:

   The status will be rolled back to the status before the upgrade. That is, file X in NameNode will have A bytes, but block files in DataNode will have A + B bytes.

Recovery procedure:

1. Obtain the list of damaged files from NameNode web UI or run the following command to obtain:

   hdfs fsck <filepath> -list-corruptfileblocks

2. Run the following command to delete unnecessary files:

   hdfs fsck <corrupt file path> - delete

   NOTE

   Deleting a file is a high-risk operation. Ensure that the files are no longer needed before performing this operation.

3. For the required files, run the **fsck** command to obtain the block list and block sequence.

- In the block sequence table provided, use the block ID to search for the data directory in the DataNode and download the corresponding block from the DataNode.
- Write all such block files in appending mode based on the sequence to construct the original file.

  Example:

  File 1--> blk_1, blk_2, blk_3

  Create a file by combining the contents of all three block files from the same sequence.
- Delete the old file from HDFS and rewrite the new file.

12.9.33.14 Why Is "java.net.SocketException: No buffer space available" Reported When Data Is Written to HDFS

Question

Why is an "java.net.SocketException: No buffer space available" exception reported when data is written to HDFS?

This problem occurs when files are written to the HDFS. Check the error logs of the client and DataNode.

The client logs are as follows:

Figure 12-25 Client logs

```
2017-07-05 21:58:06,459 INFO  [htable-pool3-t1] ipc.AbstractRpcClient: RPC Server Kerberos principal name for service=ClientService is hbase/hadoop.hadoop123.com@HADOOP12
2017-07-05 21:58:06,893 WARN  [main] mapreduce.LoadIncrementalHFiles: Skipping non-directory hdfs://hacluster/HBaseTest/bulkload_output/_SUCCESS
2017-07-05 21:59:13,211 WARN  [main] hdfs.BlockReaderFactory: I/O error constructing remote block reader.
java.net.SocketException: No buffer space available
        at sun.nio.ch.Net.connect0(Native Method)
        at sun.nio.ch.Net.connect(Net.java:454)
        at sun.nio.ch.Net.connect(Net.java:446)
        at sun.nio.ch.SocketChannelImpl.connect(SocketChannelImpl.java:648)
        at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:192)
        at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:531)
        at org.apache.hadoop.hdfs.DFSClient.newConnectedPeer(DFSClient.java:3345)
        at org.apache.hadoop.hdfs.BlockReaderFactory.nextTcpPeer(BlockReaderFactory.java:789)
        at org.apache.hadoop.hdfs.BlockReaderFactory.getRemoteBlockReaderFromTcp(BlockReaderFactory.java:706)
        at org.apache.hadoop.hdfs.BlockReaderFactory.build(BlockReaderFactory.java:369)
        at org.apache.hadoop.hdfs.DFSInputStream.getBlockReader(DFSInputStream.java:713)
        at org.apache.hadoop.hdfs.DFSInputStream.blockSeekTo(DFSInputStream.java:663)
        at org.apache.hadoop.hdfs.DFSInputStream.readWithStrategy(DFSInputStream.java:919)
        at org.apache.hadoop.hdfs.DFSInputStream.read(DFSInputStream.java:973)
        at java.io.DataInputStream.readFully(DataInputStream.java:195)
        at org.apache.hadoop.hbase.io.hfile.FixedFileTrailer.readFromStream(FixedFileTrailer.java:391)
        at org.apache.hadoop.hbase.io.hfile.HFile.isHFileFormat(HFile.java:578)
        at org.apache.hadoop.hbase.io.hfile.HFile.isHFileFormat(HFile.java:560)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.visitBulkHFiles(LoadIncrementalHFiles.java:229)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.discoverLoadQueue(LoadIncrementalHFiles.java:281)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.prepareHFileQueue(LoadIncrementalHFiles.java:452)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.doBulkLoad(LoadIncrementalHFiles.java:365)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.doBulkLoad(LoadIncrementalHFiles.java:331)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.run(LoadIncrementalHFiles.java:1107)
        at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:70)
        at org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles.main(LoadIncrementalHFiles.java:1114)
2017-07-05 21:59:13,215 WARN  [main] hdfs.DFSClient: Failed to connect to /192.168.152.128:25009 for block BP-1989348819-192.168.199.5-1497961637591:blk_1107301222_335745
ffer space available
java.net.SocketException: No buffer space available
        at sun.nio.ch.Net.connect0(Native Method)
        at sun.nio.ch.Net.connect(Net.java:454)
        at sun.nio.ch.Net.connect(Net.java:446)
        at sun.nio.ch.SocketChannelImpl.connect(SocketChannelImpl.java:648)
        at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:192)
        at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:531)
        at org.apache.hadoop.hdfs.DFSClient.newConnectedPeer(DFSClient.java:3345)
```

DataNode logs are as follows:

```
2017-07-24 20:43:39,269 | ERROR | DataXceiver for client DFSClient_NONMAPREDUCE_996005058_86
 at /192.168.164.155:40214 [Receiving block
BP-1287143557-192.168.199.6-1500707719940:blk_1074269754_528941 with io weight 10] |
DataNode{data=FSDataset{dirpath='[/srv/BigData/hadoop/data1/dn/current, /srv/BigData/hadoop/
data2/dn/current, /srv/BigData/hadoop/data3/dn/current, /srv/BigData/hadoop/data4/dn/current, /srv/
BigData/hadoop/data5/dn/current, /srv/BigData/hadoop/data6/dn/current, /srv/BigData/hadoop/data7/dn/
current]'}, localName='192-168-164-155:9866', datanodeUuid='a013e29c-4e72-400c-bc7b-bbbf0799604c',
xmitsInProgress=0}:Exception transfering block
BP-1287143557-192.168.199.6-1500707719940:blk_1074269754_528941 to mirror 192.168.202.99:9866:
java.net.SocketException: No buffer space available | DataXceiver.java:870
```

```
2017-07-24 20:43:39,269 | INFO | DataXceiver for client DFSClient_NONMAPREDUCE_996005058_86
 at /192.168.164.155:40214 [Receiving block
BP-1287143557-192.168.199.6-1500707719940:blk_1074269754_528941 with io weight 10] | opWriteBlock
BP-1287143557-192.168.199.6-1500707719940:blk_1074269754_528941 received exception
java.net.SocketException: No buffer space available | DataXceiver.java:933
2017-07-24 20:43:39,270 | ERROR | DataXceiver for client DFSClient_NONMAPREDUCE_996005058_86
 at /192.168.164.155:40214 [Receiving block
BP-1287143557-192.168.199.6-1500707719940:blk_1074269754_528941 with io weight 10] |
192-168-164-155:9866:DataXceiver error processing WRITE_BLOCK operation src: /192.168.164.155:40214
dst: /192.168.164.155:9866 | DataXceiver.java:304 java.net.SocketException: No buffer space available
 at sun.nio.ch.Net.connect0(Native Method)
 at sun.nio.ch.Net.connect(Net.java:454)
 at sun.nio.ch.Net.connect(Net.java:446)
 at sun.nio.ch.SocketChannelImpl.connect(SocketChannelImpl.java:648)
 at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:192)
 at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:531)
 at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:495)
 at org.apache.hadoop.hdfs.server.datanode.DataXceiver.writeBlock(DataXceiver.java:800)
 at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.opWriteBlock(Receiver.java:138)
 at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.processOp(Receiver.java:74)
 at org.apache.hadoop.hdfs.server.datanode.DataXceiver.run(DataXceiver.java:265)
 at java.lang.Thread.run(Thread.java:748)
```

Answer

The preceding problem may be caused by network memory exhaustion.

You can increase the threshold of the network device based on the actual scenario.

Example:

```
[root@xxxxx ~]# cat /proc/sys/net/ipv4/neigh/default/gc_thresh*
128
512
1024
[root@xxxxx ~]# echo 512 > /proc/sys/net/ipv4/neigh/default/gc_thresh1
[root@xxxxx ~]# echo 2048 > /proc/sys/net/ipv4/neigh/default/gc_thresh2
[root@xxxxx ~]# echo 4096 > /proc/sys/net/ipv4/neigh/default/gc_thresh3
[root@xxxxx ~]# cat /proc/sys/net/ipv4/neigh/default/gc_thresh*
512
2048
4096
```

You can also add the following parameters to the **/etc/sysctl.conf** file. The configuration takes effect even if the host is restarted.

```
net.ipv4.neigh.default.gc_thresh1 = 512
net.ipv4.neigh.default.gc_thresh2 = 2048
net.ipv4.neigh.default.gc_thresh3 = 4096
```

12.9.33.15 Why are There Two Standby NameNodes After the active NameNode Is Restarted?

Question

Why are there two standby NameNodes after the active NameNode is restarted?

When this problem occurs, check the ZooKeeper and ZooKeeper FC logs. You can find that the sessions used for the communication between the ZooKeeper server and client (ZKFC) are inconsistent. The session ID of the ZooKeeper server is **0x164cb2b3e4b36ae4**, and the session ID of the ZooKeeper FC is **0x144cb2b3e4b36ae4**. Such inconsistency means that the data interaction between the ZooKeeper server and ZKFC fails.

Content of the ZooKeeper log is as follows:

```
2015-04-15 21:24:54,257 | INFO | CommitProcessor:22 | Established session 0x164cb2b3e4b36ae4 with negotiated timeout 45000 for client /192.168.0.117:44586 | org.apache.zookeeper.server.ZooKeeperServer.finishSessionInit(ZooKeeperServer.java:623)
2015-04-15 21:24:54,261 | INFO | NIOServerCxn.Factory:192-168-0-114/192.168.0.114:2181 | Successfully authenticated client: authenticationID=hdfs/hadoop@<System domain name>; authorizationID=hdfs/hadoop@<System domain name>. | org.apache.zookeeper.server.auth.SaslServerCallbackHandler.handleAuthorizeCallback(SaslServerCallbackHandler.java:118)
2015-04-15 21:24:54,261 | INFO | NIOServerCxn.Factory:192-168-0-114/192.168.0.114:2181 | Setting authorizedID: hdfs/hadoop@<System domain name> | org.apache.zookeeper.server.auth.SaslServerCallbackHandler.handleAuthorizeCallback(SaslServerCallbackHandler.java:134)
2015-04-15 21:24:54,261 | INFO | NIOServerCxn.Factory:192-168-0-114/192.168.0.114:2181 | adding SASL authorization for authorizationID: hdfs/hadoop@<System domain name> | org.apache.zookeeper.server.ZooKeeperServer.processSasl(ZooKeeperServer.java:1009)
2015-04-15 21:24:54,262 | INFO | ProcessThread(sid:22 cport:-1): | Got user-level KeeperException when processing sessionid:0x164cb2b3e4b36ae4 type:create cxid:0x3 zxid:0x20009fafc txntype:-1 reqpath:n/a Error Path:/hadoop-ha/hacluster/ActiveStandbyElectorLock Error:KeeperErrorCode = NodeExists for /hadoop-ha/hacluster/ActiveStandbyElectorLock | org.apache.zookeeper.server.PrepRequestProcessor.pRequest(PrepRequestProcessor.java:648)
```

Content of the ZKFC log is as follows:

```
2015-04-15 21:24:54,237 | INFO | main-SendThread(192-168-0-114:2181) | Socket connection established to 192-168-0-114/192.168.0.114:2181, initiating session | org.apache.zookeeper.ClientCnxn$SendThread.primeConnection(ClientCnxn.java:854)
2015-04-15 21:24:54,257 | INFO | main-SendThread(192-168-0-114:2181) | Session establishment complete on server 192-168-0-114/192.168.0.114:2181, sessionid = 0x144cb2b3e4b36ae4 , negotiated timeout = 45000 | org.apache.zookeeper.ClientCnxn$SendThread.onConnected(ClientCnxn.java:1259)
2015-04-15 21:24:54,260 | INFO | main-EventThread | EventThread shut down | org.apache.zookeeper.ClientCnxn$EventThread.run(ClientCnxn.java:512)
2015-04-15 21:24:54,262 | INFO | main-EventThread | Session connected. | org.apache.hadoop.ha.ActiveStandbyElector.processWatchEvent(ActiveStandbyElector.java:547)
2015-04-15 21:24:54,264 | INFO | main-EventThread | Successfully authenticated to ZooKeeper using SASL. | org.apache.hadoop.ha.ActiveStandbyElector.processWatchEvent(ActiveStandbyElector.java:573)
```

Answer

- Cause Analysis

  After the active NameNode restarts, the temporary node **/hadoop-ha/hacluster/ActiveStandbyElectorLock** created on ZooKeeper is deleted. After the standby NameNode receives that information that the **/hadoop-ha/hacluster/ActiveStandbyElectorLock** node is deleted, the standby NameNode creates the /**hadoop-ha/hacluster/ActiveStandbyElectorLock** node in ZooKeeper in order to switch to the active NameNode. However, when the standby NameNode connects with ZooKeeper through the client ZKFC, the session ID of ZKFC differs from that of ZooKeeper due to network issues, overload CPU, or overload clusters. In this case, the watcher of the standby NameNode fails to detect that the temporary node has been successfully created, and fails to consider the standby NameNode as the active NameNode. After the original active NameNode restarts, it detects that the **/hadoop-ha/hacluster/ActiveStandbyElectorLock** already exists and becomes the standby NameNode. Therefore, both NameNodes are standby NameNodes.

- Solution

  You are advised to restart two ZKFCs of HDFS on FusionInsight Manager.

12.9.33.16 When Does a Balance Process in HDFS, Shut Down and Fail to be Executed Again?

Question

After I start a Balance process in HDFS, the process is shut down abnormally. If I attempt to execute the Balance process again, it fails again.

Answer

After a Balance process is executed in HDFS, another Balance process can be executed only after the **/system/balancer.id** file is automatically released.

However, if a Balance process is shut down abnormally, the **/system/balancer.id** has not been released when the Balance is executed again, which triggers the **append /system/balancer.id** operation.

- If the time spent on releasing the **/system/balancer.id** file exceeds the soft-limit lease period 60 seconds, executing the Balance process again triggers the append operation, which preempts the lease. The last block is in construction or under recovery status, which triggers the block recovery operation. The **/system/balancer.id** file cannot be closed until the block recovery completes. Therefore, the append operation fails.

  After the **append /system/balancer.id** operation fails, the exception message **RecoveryInProgressException** is displayed.

  ```
  org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.protocol.RecoveryInProgressException): Failed to APPEND_FILE /system/balancer.id for DFSClient because lease recovery is in progress. Try again later.
  ```

- If the time spent on releasing the **/system/balancer.id** file is within 60 seconds, the original client continues to own the lease and the exception AlreadyBeingCreatedException occurs and null is returned to the client. The following exception message is displayed on the client:

  ```
  java.io.IOException: Cannot create any NameNode Connectors.. Exiting...
  ```

Either of the following methods can be used to solve the problem:

- Execute the Balance process again after the hard-limit lease period expires for 1 hour, when the original client has released the lease.
- Delete the **/system/balancer.id** file before executing the Balance process again.

12.9.33.17 "This page can't be displayed" Is Displayed When Internet Explorer Fails to Access the Native HDFS UI

Question

Occasionally, nternet Explorer 9, Explorer 10, or Explorer 11 fails to access the native HDFS UI.

Symptom

Internet Explorer 9, Explorer 10, or Explorer 11 fails to access the native HDFS UI, as shown in the following figure.

Cause

Some Internet Explorer 9, Explorer 10, or Explorer 11versions fail to handle SSL handshake issues, causing access failure.

Solution

Refresh the page.

12.9.33.18 NameNode Fails to Be Restarted Due to EditLog Discontinuity

Question

If a JournalNode server is powered off, the data directory disk is fully occupied, and the network is abnormal, the EditLog sequence number on the JournalNode is inconsecutive. In this case, the NameNode restart may fail.

Symptom

The NameNode fails to be restarted. The following error information is reported in the NameNode run logs:

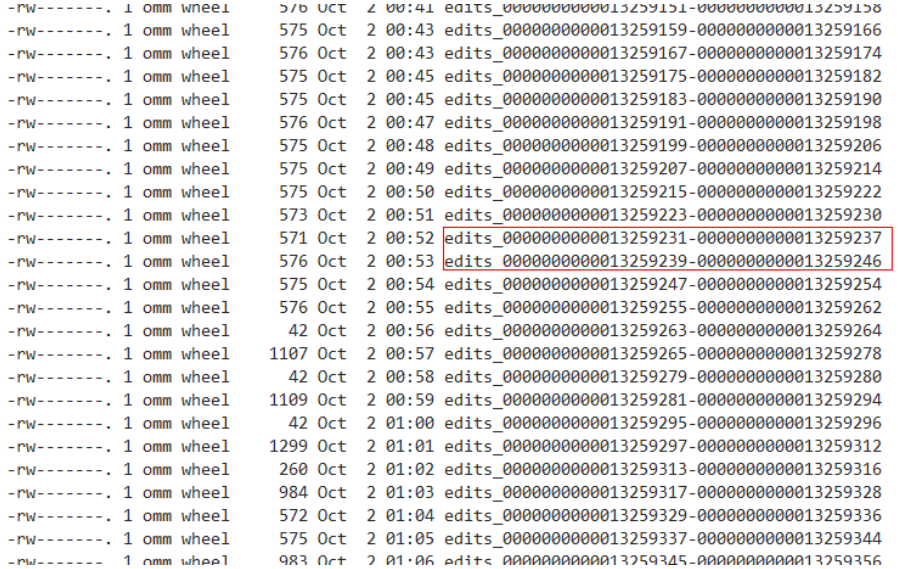
```
2019-11-08 16:30:28,399 | ERROR | main | Failed to start namenode. | NameNode.java:1732
java.io.IOException: There appears to be a gap in the edit log.  We expected txid 13698019, but got txid 13698088.
        at org.apache.hadoop.hdfs.server.namenode.MetaRecoveryContext.editLogLoaderPrompt(MetaRecoveryContext.java:94)
        at org.apache.hadoop.hdfs.server.namenode.FSEditLogLoader.loadEditRecords(FSEditLogLoader.java:278)
        at org.apache.hadoop.hdfs.server.namenode.FSEditLogLoader.loadFSEdits(FSEditLogLoader.java:188)
        at org.apache.hadoop.hdfs.server.namenode.FSImage.loadEdits(FSImage.java:924)
        at org.apache.hadoop.hdfs.server.namenode.FSImage.loadFSImage(FSImage.java:771)
        at org.apache.hadoop.hdfs.server.namenode.FSImage.recoverTransitionRead(FSImage.java:331)
        at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFSImage(FSNamesystem.java:1108)
        at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFromDisk(FSNamesystem.java:727)
        at org.apache.hadoop.hdfs.server.namenode.NameNode.loadNamesystem(NameNode.java:638)
        at org.apache.hadoop.hdfs.server.namenode.NameNode.initialize(NameNode.java:700)
        at org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.java:943)
        at org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.java:916)
        at org.apache.hadoop.hdfs.server.namenode.NameNode.createNameNode(NameNode.java:1655)
        at org.apache.hadoop.hdfs.server.namenode.NameNode.main(NameNode.java:1725)
```

Solution

1. Find the active NameNode before the restart, go to its data directory (you can obtain the directory, such as **/srv/BigData/namenode/current** by checking the configuration item **dfs.namenode.name.dir**), and obtain the sequence number of the latest FsImage file, as shown in the following figure:

```
-rw-------. 1 omm wheel     574 Oct  2 01:12 edits_0000000000013259401-00000000000
-rw-------. 1 omm wheel     575 Oct  2 01:13 edits_0000000000013259409-00000000000
-rw-------. 1 omm wheel      42 Oct  2 01:13 edits_0000000000013259417-00000000000
-rw-------. 1 omm wheel 1048576 Nov  8 16:01 edits_inprogress_0000000000013698088
-rw-------. 1 omm wheel  314803 Nov  8 15:53 fsimage_0000000000013698018
-rw-------. 1 omm wheel      62 Nov  8 15:53 fsimage_0000000000013698018.md5
-rw-------. 1 omm wheel  314803 Nov  8 15:56 fsimage_0000000000013698050
-rw-------. 1 omm wheel      62 Nov  8 15:56 fsimage_0000000000013698050.md5
-rw-------. 1 omm wheel  314803 Nov  8 15:59 fsimage_0000000000013698066
-rw-------. 1 omm wheel      62 Nov  8 15:59 fsimage_0000000000013698066.md5
-rw-------. 1 omm wheel       9 Oct  2 01:13 seen_txid
-rw-------. 1 omm wheel     187 Nov  8 15:59 VERSION
```

2. Check the data directory of each JournalNode (you can obtain the directory such as**/srv/BigData/journalnode/hacluster/current** by checking the value of the configuration item **dfs.journalnode.edits.dir**), and check whether the sequence number starting from that obtained in step 1 is consecutive in edits files. That is, you need to check whether the last sequence number of the previous edits file is consecutive with the first sequence number of the next edits file. (As shown in the following figure, edits_0000000000013259231-0000000000013259237 and edits_0000000000013259239-0000000000013259246 are not consecutive.)



3. If the edits files are not consecutive, check whether the edits files with the related sequence number exist in the data directories of other JournalNodes or NameNode. If the edits files can be found, copy a consecutive segment to the JournalNode.
4. In this way, all inconsecutive edits files are restored.
5. Restart the NameNode and check whether the restart is successful. If the fault persists, contact technical support.

12.10 Using Hive

12.10.1 Using Hive from Scratch

Hive is a data warehouse framework built on Hadoop. It maps structured data files to a database table and provides SQL-like functions to analyze and process data. It also allows you to quickly perform simple MapReduce statistics using SQL-like statements without the need of developing a specific MapReduce application. It is suitable for statistical analysis of data warehouses.

Background

Suppose a user develops an application to manage users who use service A in an enterprise. The procedure of operating service A on the Hive client is as follows:

Operations on common tables:

- Create the **user_info** table.
- Add users' educational backgrounds and professional titles to the table.
- Query user names and addresses by user ID.
- Delete the user information table after service A ends.

Table 12-227 User information

ID	Name	Gender	Age	Address
12005000201	A	Male	19	City A
12005000202	B	Female	23	City B
12005000203	C	Male	26	City C
12005000204	D	Male	18	City D
12005000205	E	Female	21	City E
12005000206	F	Male	32	City F
12005000207	G	Female	29	City G
12005000208	H	Female	30	City H
12005000209	I	Male	26	City I
12005000210	J	Female	25	City J

Procedure

Step 1 Download the client configuration file.

- For versions earlier than MRS 3.x, perform the following operations:
  a. Log in to MRS Manager. For details, see **Accessing Manager**. Then, choose **Services**.
  b. Click **Download Client**.
     Set **Client Type** to **Only configuration files**, **Download to** to **Server**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default.
- For MRS 3.*x* or later, perform the following operations:
  a. Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.
  b. Choose **Cluster** > *Name of the desired cluster* > **Dashboard** > **More** > **Download Client**.
  c. Download the cluster client.
     Set **Select Client Type** to **Configuration Files Only** , select a platform type, and click **OK** to generate the client configuration file which is then saved in the **/tmp/FusionInsight-Client/** directory on the active management node by default.

Step 2 Log in to the active management node of Manager.

- For versions earlier than MRS 3.x, perform the following operations:
  a. On the MRS console, click **Clusters**, choose **Active Clusters**, and click a cluster name. On the **Nodes** tab, view the node names. The node whose name contains **master1** is the Master1 node, and the node whose name contains **master2** is the Master2 node.

     The active and standby management nodes of MRS Manager are installed on Master nodes by default. Because Master1 and Master2 are switched over in active and standby mode, Master1 is not always the active management node of MRS Manager. Run a command in Master1 to check whether Master1 is active management node of MRS Manager. For details about the command, see **Step 2.d**.
  b. Log in to the Master1 node using the password as user **root**.
  c. Run the following commands to switch to user **omm**:

     sudo su - root

     su - omm
  d. Run the following command to check the active management node of MRS Manager:

     sh ${BIGDATA_HOME}/om-0.0.1/sbin/status-oms.sh

     In the command output, the node whose **HAActive** is **active** is the active management node, and the node whose **HAActive** is **standby** is the standby management node. In the following example, **mgtomsdat-sh-3-01-1** is the active management node, and **mgtomsdat-sh-3-01-2** is the standby management node.

```
Ha mode
double
NodeName            HostName                HAVersion         StartTime
HAActive            HAAllResOK          HARunPhase
192-168-0-30        mgtomsdat-sh-3-01-1     V100R001C01       2014-11-18 23:43:02
active              normal              Actived
192-168-0-24        mgtomsdat-sh-3-01-2     V100R001C01       2014-11-21 07:14:02
standby             normal              Deactived
```

  e. Log in to the active management node as user **root**, for example, node **192-168-0-30**.
- For MRS 3.*x* or later, perform the following operations:
  a. Log in to any node where Manager is deployed as user **root**.
  b. Run the following command to identify the active and standby nodes:

     sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh

     In the command output, the value of **HAActive** for the active management node is **active**, and that for the standby management node is **standby**. In the following example, **node-master1** is the active management node, and **node-master2** is the standby management node.

```
HAMode
double
NodeName            HostName        HAVersion         StartTime             HAActive
HAAllResOK          HARunPhase
192-168-0-30        node-master1    V100R001C01       2020-05-01 23:43:02   active
normal              Actived
192-168-0-24        node-master2    V100R001C01       2020-05-01 07:14:02
standby             normal          Deactived
```

c. Log in to the primary management node as user **root** and run the following command to switch to user **omm**:

sudo su - omm

Step 3 Run the following command to go to the client installation directory:

cd /opt/client

The cluster client has been installed in advance. The following client installation directory is used as an example. Change it based on the site requirements.

Step 4 Run the following command to update the client configuration for the active management node.

sh refreshConfig.sh /opt/client *Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/client /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully.

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

Step 5 Use the client on a Master node.

1. On the active management node, for example, **192-168-0-30**, run the following command to switch to the client directory, for example, **/opt/client**.

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user:

   kinit *MRS cluster user*

   Example: user **kinit hiveuser**

   The current user must have the permission to create Hive tables.If Kerberos authentication is disabled, skip this step.

4. Run the client command of the Hive component directly.

   beeline

Step 6 Run the Hive client command to implement service A.

Operations on internal tables:

1. Create the **user_info** user information table according to **Table 12-227** and add data to it.

   create table user_info(id string,name string,gender string,age int,addr string);

   For MRS 1.*x*, MRS 3.*x*, or later, perform the following operations:

   insert into table user_info(id,name,gender,age,addr) values("12005000201","A","Male",19,"City A");

   For MRS 2.*x*, perform the following operations:

insert into table user_info values("12005000201","A","Male",19,"City A");

2. Add users' educational backgrounds and professional titles to the **user_info** table.

   For example, to add educational background and title information about user 12005000201, run the following command:

   alter table user_info add columns(education string,technical string);

3. Query user names and addresses by user ID.

   For example, to query the name and address of user 12005000201, run the following command:

   select name,addr from user_info where id='12005000201';

4. Delete the user information table.

   drop table user_info;

Operations on external partition tables:

Create an external partition table and import data.

1. Create a path for storing external table data.

   hdfs dfs -mkdir /hive/

   hdfs dfs -mkdir /hive/user_info

2. Create a table.

   create external table user_info(id string,name string,gender string,age int,addr string) partitioned by(year string) row format delimited fields terminated by ' ' lines terminated by '\n' stored as textfile location '/hive/user_info';

   NOTE

   fields terminated indicates delimiters, for example, spaces.

   lines terminated indicates line breaks, for example, **\n**.

   /hive/user_info indicates the path of the data file.

3. Import data.

   a. Execute the insert statement to insert data.

      insert into user_info partition(year="2018") values ("12005000201","A","Male",19,"City A");

   b. Run the **load data** command to import file data.

      i. Create a file based on the data in **Table 12-227**. For example, the file name is **txt.log**. Fields are separated by space, and the line feed characters are used as the line breaks.

      ii. Upload the file to HDFS.

      hdfs dfs -put txt.log /tmp

      iii. Load data to the table.

      load data inpath '/tmp/txt.log' into table user_info partition (year='2011');

4. Query the imported data.

   select * from user_info;

5. Delete the user information table.

   drop table user_info;

6. Run the following command to exit:

   !q

----End

12.10.2 Configuring Hive Parameters

Navigation Path

Go to the Hive configurations page by referring to **Modifying Cluster Service Configuration Parameters**.

Parameter Description

Table 12-228 Hive parameter description

Parameter	Description	Default Value
hive.auto.convert.join	Whether Hive converts common **join** to **mapjoin** based on the input file size. **NOTE** When Hive is used to query a join table, whatever the table size is (if the data in the join table is less than 24 MB, it is a small one), set this parameter to **false**. If this parameter is set to **true**, new **mapjoin** cannot be generated when you query a join table.	Possible values are as follows: • true • false The default value is **true**.
hive.default.fileformat	Indicates the default file format used by Hive.	Versions earlier than MRS 3.x: TextFile MRS 3.x or later: RCFile
hive.exec.reducers.max	Indicates the maximum number of reducers in a MapReduce job submitted by Hive.	999
hive.server2.thrift.max.worker.threads	Indicates the maximum number of threads that can be started in the HiveServer internal thread pool.	1,000
hive.server2.thrift.min.worker.threads	Indicates the number of threads started during initialization in the HiveServer internal thread pool.	5

Parameter	Description	Default Value
hive.hbase.delete.mode.enabled	Indicates whether to enable the function of deleting HBase records from Hive. If this function is enabled, you can use **remove table xx where xxx** to delete HBase records from Hive. **NOTE** This parameter applies to MRS 3.*x* or later.	true
hive.metastore.server.min.threads	Indicates the number of threads started by MetaStore for processing connections. If the number of threads is more than the set value, MetaStore always maintains a number of threads that is not lower than the set value, that is, the number of resident threads in the MetaStore thread pool is always higher than the set value.	200
hive.server2.enable.doAs	Indicates whether to simulate client users during sessions between HiveServer2 and other services (such as Yarn and HDFS). If you change the configuration item from **false** to **true**, users with only the column permission lose the permissions to access corresponding tables. **NOTE** This parameter applies to MRS 3.*x* or later.	true

12.10.3 Hive SQL

Hive SQL supports all features of Hive-3.1.0. For details, see **https://cwiki.apache.org/confluence/display/hive/languagemanual**.

Table 12-229 describes the extended Hive statements provided by .

Table 12-229 Extended Hive statements

Extended Syntax	Syntax Description	Syntax Example	Example Description
CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS] [db_name.]table_name (col_name data_type [COMMENT col_comment], ...) [ROW FORMAT row_format] [STORED AS file_format] | STORED BY 'storage.handler.class.name' [WITH SERDEPROPERTIES (...)] ..... [TBLPROPERTIES ("groupId"=" group1 ","locatorId"="locator1")] ...;	The statement is used to create a Hive table and specify locators on which table data files locate. For details, see **Using HDFS Colocation to Store Hive Tables**.	CREATE TABLE tab1 (id INT, name STRING) row format delimited fields terminated by '\t' stored as RCFILE TBLPROPERTIES(" groupId"=" group1 ","locatorId"="locator1");	The statement is used to create table **tab1** and specify locator1 on which the table data of **tab1** locates.

Extended Syntax	Syntax Description	Syntax Example	Example Description
CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS] [db_name.]table_name (col_name data_type [COMMENT col_comment], ...) [ROW FORMAT row_format] [STORED AS file_format] | STORED BY 'storage.handler.class.name' [WITH SERDEPROPERTIES (...)] ... [TBLPROPERTIES ('**column.encode.columns**'='col_name1,col_name2'| '**column.encode.indices**'='col_id1,col_id2', **'column.encode.classname'**='encode_classname')]...;	The statement is used to create a hive table and specify the table encryption column and encryption algorithm. For details, see **Using the Hive Column Encryption Function**.	create table encode_test(id INT, name STRING, phone STRING, address STRING) ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe' WITH SERDEPROPERTIES ('column.encode.indices'='2,3', 'column.encode.classname'='org.apache.hadoop.hive.serde2.SMS4Rewriter') STORED AS TEXTFILE;	The statement is used to create table **encode_test** and specify that column 2 and column 3 will be encrypted using the **org.apache.hadoop.hive.serde2.SMS4Rewriter** encryption algorithm class during data insertion.
REMOVE TABLE *hbase_tablename* [WHERE where_condition];	The statement is used to delete data that meets criteria from the Hive on HBase table. For details, see **Deleting Single-Row Records from Hive on HBase**.	remove table hbase_table1 where id = 1;	The statement is used to delete data that meets the criterion of "id = 1" from the table.

Extended Syntax	Syntax Description	Syntax Example	Example Description
CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS] [db_name.]table_name (col_name data_type [COMMENT col_comment], ...) [ROW FORMAT row_format] **STORED AS inputformat 'org.apache.hadoop.hive.contrib.fileformat.SpecifiedDelimiterInputFormat'** outputformat 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat';	The statement is used to create a hive table and specify that the table supports customized row delimiters. For details, see **Customizing Row Separators**.	create table blu(time string, num string, msg string) row format delimited fields terminated by ',' **stored as inputformat 'org.apache.hadoop.hive.contrib.fileformat.SpecifiedDelimiterInputFormat'** outputformat 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat';	The statement is used to create table **blu** and set **inputformat** to **SpecifiedDelimiterInputFormat** so that the query row delimiter can be specified during the query.

12.10.4 Permission Management

12.10.4.1 Hive Permission

Hive is a data warehouse framework built on Hadoop. It provides basic data analysis services using the Hive query language (HQL), a language like the structured query language (SQL).

MRS supports users, user groups, and roles. Permissions must be assigned to roles and then roles are bound to users or user groups. Users can obtain permissions only by binding a role or joining a group that is bound with a role. For details about Hive authorization, visit **https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Authorization**.

NOTE

- Hive permissions in security mode need to be managed whereas those in normal mode do not.
- MRS 3.*x* or later supports Ranger. If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Hive**.

Hive Permission Model

To use the Hive component, users must have permissions on Hive databases and tables (including external tables and views). In MRS, the complete Hive permission model is composed of Hive metadata permission and HDFS file permission. The Hive permission model also includes the permission to use databases or tables.

- Hive metadata permission

  Similar to traditional relational databases, the Hive database of MRS supports the **CREATE** and **SELECT** permission, and the Hive tables and columns support the **SELECT**, **INSERT**, and **DELETE** permissions. Hive also supports the permissions of **OWNERSHIP** and **Hive Admin Privilege**.

  NOTE

  The **UPDATE** and **DELETE** operations on Hive tables and columns can be performed only when **ACID** is enabled. **ACID** cannot be enabled in the current version.

- Hive data file permission, also known as HDFS file permission

  Hive database and table files are stored in the HDFS. The created databases or tables are saved in the **/user/hive/warehouse** directory of the HDFS by default. The system automatically creates subdirectories named after database names and database table names. To access a database or a table, the corresponding file permissions (read, write, and execute) on the HDFS are required.

  NOTE

  MRS 3.*X* supports multiple Hive instances. In the multi-instance scenario, the directory is **/user/hiven** *n* (*n*=1–4)**/warehouse**.

To perform various operations on Hive databases or tables, you need to associate the metadata permission with the HDFS file permission. For example, to query Hive data tables, you need to associate the metadata permission **SELECT** and the HDFS file permissions **Read** and **Write**.

To use the role management function of Manager GUI to manage the permissions of Hive databases and tables, you only need to configure the metadata permission, and the system will automatically associate and configure the HDFS file permission. In this way, operations on the interface are simplified, and the efficiency is improved.

Hive Users

MRS provides users and roles to use Hive, such as creating tables, inserting data into tables, and querying tables. Hive defines the **USER** class, corresponding to user instances. Hive defines the **GROUP** class, corresponding to role instances.

You can use Manager to set permissions for Hive users. This method only supports permission setting in roles. A user or user group can obtain the permissions only after a role is bound to the user or user group. Hive users can be granted administrator permissions and permissions to access databases, tables, and columns.

Hive Usage Scenarios and Related Permissions

Creating a database with Hive requires users to join in the **hive** group, without granting a role. Users have all permissions on the databases or tables created by

themselves in Hive or HDFS. They can create tables, select, delete, insert, or update data, and grant permissions to other users to allow them to access the tables and corresponding HDFS directories and files.

A user can access the tables or database only with permissions. The permission required by users varies according to Hive usage scenarios.

Table 12-230 Hive usage scenarios

Typical Scenario	Permission
Using Hive tables, columns, or databases	Permissions required in different scenarios are as follows: ● To create tables, the **CREATE** permission is required. ● To query data, the **SELECT** permission is required. ● To insert data, the **INSERT** permission is required. ● To delete data, the **DELETE** permission is required.
Associating and using other components	In addition to Hive permissions, permissions of other components are required in some scenarios, for example: ● Yarn permissions are required when some HQL statements, such as **insert**, **count**, **distinct**, **group by**, **order by**, **sort by**, and **join**, are run. You are advised to grant Yarn permissions to the role of each Hive user. ● HBase permission is required when Hive over HBase is used, for example, querying HBase table data in Hive.

In some special Hive usage scenarios, you need to configure other types of permission.

Table 12-231 Hive authorization precautions

Scenario	Permission
Creating Hive databases, tables, and external tables, or adding partitions to created Hive tables or external tables when data files specified by Hive users are saved to other HDFS directories except **/user/hive/warehouse**	The directory must already exist, the Hive user must be the owner of the directory, and the Hive user must have the read, write, and execute permissions on the directory. The user must have the **read** and **write** permissions of all the upper-layer directories of the directory. After an administrator grants the Hive permission to the role, the HDFS permission is automatically granted.
Using **load** to load data from all the files or specified files in a specified directory to Hive tables as a Hive user	• The data source is a Linux local disk, the specified directory exists, and the system user **omm** has read and execute permission of the directory and all its upper-layer directories. The specified file exists, and user **omm** has read permission of the file and has the read and execute permission of all the upper-layer directories of the file. • The data source is HDFS, the specified directory exists, and the Hive user is the owner of the directory and has read, write, and execute permission on the directory and its subdirectories, and has read and write permission on all its upper-layer directories. The specified file exists, and the Hive user is the owner of the file and has read, write, and execute permission, and has read and execute permission on the file and all its upper-layer directories. **NOTE** When **load** is used to import data to a Linux local disk, files must be loaded to the HiveServer on which the command is run and the permission must be modified. You are advised to run the command on a client. The HiveSever to which the client is connected can be found. For example, if the Hive client displays **0: jdbc:hive2://10.172.0.43:21066/>**, the IP address of the connected HiveServer is 10.172.0.43.
Creating or deleting functions or modifying any database	The **Hive Admin Privilege** is required.

Scenario	Permission
Performing operations on all databases and tables in Hive	The user must be added to the **supergroup** user group and granted **Hive Admin Privilege**.

12.10.4.2 Creating a Hive Role

Scenario

This section describes how to create and configure a Hive role on Manager as the system administrator. The Hive role can be granted the permissions of the Hive administrator and the permissions to operate Hive table data.

Creating a database with Hive requires users to join in the **hive** group, without granting a role. Users have all permissions on the databases or tables created by themselves in Hive or HDFS. They can create tables, select, delete, insert, or update data, and grant permissions to other users to allow them to access the tables and corresponding HDFS directories and files. The created databases or tables are saved in the **/user/hive/warehouse** directory of the HDFS by default.

NOTE

- A Hive role can be created only in security mode.
- If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Hive** for MRS 3.*x* or later that supports Ranger.

Prerequisites

- The system administrator has understood the service requirements.
- Log in to FusionInsight Manager.
- The Hive client has been installed.

Procedure

For versions earlier than MRS 3.x, perform the following operations to create a Hive role:

Step 1 Log in to MRS Manager.

Step 2 Choose **System** > **Permission** > **Manage Role**.

Step 3 Click **Create Role**, and set **Role Name** and **Description**.

Step 4 Set permissions. For details, see **Table 12-232**.

- **Hive Admin Privilege**: Hive administrator permissions. If you want to use this permission, run the **set role admin** command to set the permission before running SQL statements.
- **Hive Read Write Privileges**: Hive data table management permission, which is the operation permission to set and manage the data of created tables.

Select the permissions of a database as required. To specify permissions on tables, click the database name and select the permissions of the tables.

NOTE

- Hive role management supports the administrator permission, and the permissions of accessing tables and views, without granting the database permission.
- The permissions of the Hive administrator do not include the permission to manage HDFS.
- If there are too many tables in the database or too many files in tables, the permission granting may last a while. For example, if a table contains 10,000 files, the permission granting lasts about 2 minutes.

Table 12-232 Setting a role

Scenario	Role Authorization
Setting the Hive administrator permission	In the **Permission** table, click **Hive** and select **Hive Admin Privilege**. **NOTE** After being bound to the Hive administrator role, perform the following operations during each maintenance operation: 1. Log in to the node where the client is installed. 2. Run the following command to configure environment variables: For example, if the Hive client installation directory is **/opt/hiveclient**, run **source /opt/hiveclient/bigdata_env**. 3. Run the following command to authenticate the user: **kinit** *Hive service user* 4. Run the following command to log in to the client tool: **beeline** 5. Run the following command to update the administrator permissions: **set role admin;**
Setting the permission to query a table of another user in the default database	1. In the **Permission** table, choose **Hive** > **Hive Read Write Privileges**. 2. In the **Permission** column of the specified table, select **SELECT**.
Setting the permission to query a table of another user in the default database	1. In the **Permission** table, choose **Hive** > **Hive Read Write Privileges**. 2. In the **Permission** column of the specified table, select **Insert**.

Scenario	Role Authorization
Setting the permission to import data to a table of another user in the default database	1. In the **Permission** table, choose **Hive** > **Hive Read Write Privileges**. 2. In the **Permission** column of the specified table, select **Delete** and **Insert**.
Setting the permission to submit HQL commands to Yarn for execution	The HQL commands used by some services are converted into MapReduce tasks and submitted to Yarn for execution. You need to set the Yarn permissions. For example, the HQL statements to be run use statements, such as **insert**, **count**, **distinct**, **group by**, **order by**, **sort by**, or **join**. 1. In the **Permission** table, choose **Yarn** > **Scheduler Queue** > **root**. 2. In the **Permission** column of the default queue, select **Submit**.

Step 5 Click **OK**, and return to the **Role** page.

Step 6 Choose **System** > **Manage User** > **Create User**.

Step 7 Enter the username, set **User Type** to **Human-machine**, set the user password, add a user group bound with the Hive administrator role, bind the new Hive role to the user group, and click **OK**.

Step 8 After the user is created, you can run the SQL statement using the user.

----End

For MRS 3.*x* or later, perform the following operations to create a Hive role:

Step 1 Log in to FusionInsight Manager. For details, see**Accessing FusionInsight Manager (MRS 3.x or Later)**

Step 2 Choose **System** > **Permission** > **Role**.

Step 3 Click **Create Role**, and set **Role Name** and **Description**.

Step 4 Set **Configure Resource Permission**. For details, see **Table 12-233**.

- Grant the read and execution permissions for the HDFS directory.
  - Click *Name of the desired cluster* and select **HDFS** for **Service Name**. On the displayed page, click **File System**, choose **hdfs://hacluster/** > **user**, locate the row where **hive** is located, and select **Read** and **Execute** in the **Permission** column.
  - Click *Name of the desired cluster* and select **HDFS** for **Service Name**. On the displayed page, click **File System**, choose **hdfs://hacluster/** > **user** > **hive**, locate the row where **warehouse** is located, and select **Read** and **Execute** in the **Permission** column.
  - Click *Name of the desired cluster* and select **HDFS** for **Service Name**. On the displayed page, click **File System**, choose **hdfs://hacluster/** > **tmp**,

locate the row where **hive-scratch** is located, and select **Read** and **Execute** in the **Permission** column.

- **Hive Admin Privilege**: Hive administrator permission.
- **Hive Read Write Privileges**: Hive data table management permission, which is the operation permission to set and manage the data of created tables.

NOTE

- In MRS 3.1.0, Hive role management supports the administrator permission, and the permissions of accessing tables and views, without granting the database permission.
- The permissions of the Hive administrator do not include the permission to manage HDFS.
- If there are too many tables in the database or too many files in tables, the permission granting may last a while. For example, if a table contains 10,000 files, the permission granting lasts about 2 minutes.

Table 12-233 Setting a role

Task	Role Authorization
Setting the Hive administrator permission	In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** and select **Hive Admin Privilege**. **NOTE** After being bound to the Hive administrator role, perform the following operations during each maintenance operation: 1. Log in to the node where the Hive client is installed as the client installation user. 2. Run the following command to configure environment variables: For example, if the Hive client installation directory is **/opt/hiveclient**, run **source /opt/hiveclient/bigdata_env**. 3. Run the following command to authenticate the user: **kinit** *Hive service user* 4. Run the following command to log in to the client tool: **beeline** 5. Run the following command to update the administrator permissions: **set role admin;**

Task	Role Authorization
Setting the permission to query a table of another user in the default database	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**. 2. Click the name of the specified database in the database list. Tables in the database are displayed. 3. In the **Rights** column of the specified table, choose **Select**.
Setting the permission to query a table of another user in the default database	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**. 2. Click the name of the specified database in the database list. Tables in the database are displayed. 3. In the **Permission** column of the specified table, select **INSERT**.
Setting the permission to import data to a table of another user in the default database	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**. 2. Click the name of the specified database in the database list. Tables in the database are displayed. 3. In the **Permission** column of the specified indexes, select **DELETE** and **INSERT**.
Setting the permission to submit HQL commands to Yarn for execution	The HQL commands used by some services are converted into MapReduce tasks and submitted to Yarn for execution. You need to set the Yarn permissions. For example, the HQL statements to be run use statements, such as **insert**, **count**, **distinct**, **group by**, **order by**, **sort by**, or **join**. 1. In the **Permission** table, choose *Name of the desired cluster* > **Yarn** > **Scheduling Queue** > **root**. 2. In the **Permission** column of the **default** queue, select **Submit**.

Step 5 Click **OK**, and return to the **Role** page.

----End

12.10.4.3 Configuring Permissions for Hive Tables, Columns, or Databases

Scenario

You can configure related permissions if you need to access tables or databases created by other users. Hive supports column-based permission control. If a user needs to access some columns in tables created by other users, the user must be granted the permission for columns. The following describes how to grant table, column, and database permissions to users by using the role management function of MRS Manager.

NOTE

- You can configure permissions for Hive tables, columns, or databases only in security mode.
- MRS 3.x or later supports Ranger. If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Hive**.

Prerequisites

- You have obtained a user account with the administrator permissions, such as **admin**.
- You have created a role, for example, **hrole**, on Manager by referring to instructions in **Creating a Hive Role**. You do not need to set the Hive permission but need to set the permission to submit the HQL command to Yarn for execution.
- You have created two Hive human-machine users, such as **huser1** and **huser2**, on Manager and added them to the **hive** group. **huser2** has been bound to **hrole**. The **hdb** database has created by user **huser1** and the **htable** table has been created in the database.

Procedure

- Granting Table Permissions

  Users have complete permission on the tables created by themselves in Hive and the HDFS. To access the tables created by others, they need to be granted the permission. After the Hive metadata permission is granted, the HDFS permission is automatically granted. The procedure for granting a role the permission of querying, inserting, and deleting **htable** data is as follows:

  For versions earlier than MRS 3.x, perform the following operations to grant table permissions:

  a. On MRS Manager, choose **System** > **Permission** > **Manage Role**.

  b. Locate the row that contains **hrole**, and click **Modify**.

  c. Choose **Hive** > **Hive Read Write Privileges**.

  d. Click the name of the specified database **hdb** in the database list. Table **htable** in the database is displayed.

e. In the **Permission** column of the **htable** table, select **Select**, **Insert**, and **Delete**.

f. Click **OK**.

For MRS 3.*x* or later, perform the following operations to grant table permissions:

a. On FusionInsight Manager, choose **System** > **Permission** > **Role**.

b. Locate the row that contains **hrole**, and click **Modify**.

c. Choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**.

d. Click the name of the specified database **hdb** in the database list. Table **htable** in the database is displayed.

e. In the **Permission** column of the **htable** table, select **SELECT**, **INSERT**, and **DELETE**.

f. Click **OK**.

NOTE

In role management, the procedure for granting a role the permission of querying, inserting, and deleting Hive external table data is the same. After the metadata permission is granted, the HDFS permission is automatically granted.

- Granting Column Permissions

  Users have all permissions for the tables created by themselves in Hive and HDFS. Users do not have the permission to access the tables created by others. If a user needs to access some columns in tables created by other users, the user must be granted the permission for columns. After the Hive metadata permission is granted, the HDFS permission is automatically granted. The procedure for granting a role the permission of querying and inserting data in **hcol** of **htable** is as follows:

  For versions earlier than MRS 3.x, perform the following operations to grant column permissions:

  a. On MRS Manager, choose **System** > **Permission** > **Manage Role**.

  b. Locate the row that contains **hrole**, and click **Modify**.

  c. Choose **Hive** > **Hive Read Write Privileges**.

  d. In the database list, click the specified database **hdb** to display the **htable** table in the database. Click the **htable** table to display the **hcol** column in the table.

  e. In the **Permission** column of the **hcol** column, select **Select** and **Insert**.

  f. Click **OK**.

  For MRS 3.*x* or later, perform the following operations:

  a. On FusionInsight Manager, choose **System** > **Permission** > **Role**.

  b. Locate the row that contains **hrole**, and click **Modify**.

  c. Choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**.

  d. In the database list, click the specified database **hdb** to display the **htable** table in the database. Click the **htable** table to display the **hcol** column in the table.

e. In the **Permission** column of the **hcol** column, select **SELECT** and **INSERT**.

f. Click **OK**.

NOTE

In role management, after the metadata permission is granted, the HDFS permission is automatically granted. Therefore, after the column permission is granted, the HDFS ACL permission for all files of the table is automatically granted.

- Granting Database Permissions

  Users have complete permission on the databases created by themselves in Hive and the HDFS. To access the databases created by others, they need to be granted the permission. After the Hive metadata permission is granted, the HDFS permission is automatically granted. The procedure for granting a role the permission of querying data and creating tables in database **hdb** is as follows. Other types of database operation permission are not supported.

  For versions earlier than MRS 3.x, perform the following database authorization operations:

  a. On MRS Manager, choose **System** > **Permission** > **Manage Role**.

  b. Locate the row that contains **hrole**, and click **Modify**.

  c. Choose **Hive** > **Hive Read Write Privileges**.

  d. In the **Permission** column of the **hdb** database, select **Select** and **Create**.

  e. Click **OK**.

  For MRS 3.*x* or later, perform the following operations to grant database permissions:

  a. On FusionInsight Manager, choose **System** > **Permission** > **Role**.

  b. Locate the row that contains **hrole**, and click **Modify**.

  c. Choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**.

  d. In the **Permission** column of the **hdb** database, select **SELECT** and **CREATE**.

  e. Click **OK**.

NOTE

- Any permission for a table in the database is automatically associated with the HDFS permission for the database directory to facilitate permission management. When any permission for a table is canceled, the system does not automatically cancel the HDFS permission for the database directory to ensure performance. In this case, users can only log in to the database and view table names.
- When the query permission on a database is added to or deleted from a role, the query permission on tables in the database is automatically added to or deleted from the role.

Concepts

Table 12-234 Scenarios of using Hive tables, columns, or databases

Scenario	Required Permission
DESCRIBE TABLE	SELECT

Scenario	Required Permission
SHOW PARTITIONS	SELECT
ANALYZE TABLE	SELECT and INSERT
SHOW COLUMNS	SELECT
SHOW TABLE STATUS	SELECT
SHOW TABLE PROPERTIES	SELECT
SELECT	SELECT
EXPLAIN	SELECT
CREATE VIEW	SELECT, Grant Of Select, and CREATE
SHOW CREATE TABLE	SELECT and Grant Of Select
CREATE TABLE	CREATE
ALTER TABLE ADD PARTITION	INSERT
INSERT	INSERT
INSERT OVERWRITE	INSERT and DELETE
LOAD	INSERT and DELETE
ALTER TABLE DROP PARTITION	DELETE
CREATE FUNCTION	Hive Admin Privilege
DROP FUNCTION	Hive Admin Privilege
ALTER DATABASE	Hive Admin Privilege

12.10.4.4 Configuring Permissions to Use Other Components for Hive

Scenario

Hive may need to be associated with other components. For example, Yarn permissions are required in the scenario of using HQL statements to trigger MapReduce jobs, and HBase permissions are required in the Hive over HBase scenario. The following describes the operations in the two scenarios.

NOTE

- In security mode, Yarn and HBase permission management is enabled by default. Therefore, Yarn and HBase permissions need to be configured by default.
- In common mode, Yarn and HBase permission management is disabled by default. That is, any user has permissions. Therefore, YARN and HBase permissions does not need to be configured by default. If a user enables the permission management by modifying the Yarn or HBase configurations, the Yarn and HBase permissions then need to be configured.
- MRS 3.*x* or later supports Ranger. If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Hive**.

Prerequisites

- The Hive client has been installed. For example, the installation directory is **/opt/client**.
- You have obtained a user account with the administrator permissions, such as **admin**.

Procedure

Association with Yarn in MRS Earlier than 3.x

Yarn permissions are required when HQL statements, such as **insert**, **count**, **distinct**, **group by**, **order by**, **sort by**, and **join**, are used to trigger MapReduce jobs. The following uses the procedure for assigning a role the permissions to run the **count** statements in the **thc** table as an example.

Step 1 Create a role on MRS Manager.

Step 2 In the **Permission** table, choose **Yarn** > **Scheduler Queue** > **root**.

Step 3 In the **Permission** column of the default queue, select **Submit** and click **OK**.

Step 4 In the **Permission** table, choose **Hive** > **Hive Read Write Privileges** > **default**, select **Select** for **thc**, and click **OK**.

----End

Association with Yarn in MRS 3.*x* or Later

Yarn permissions are required when HQL statements, such as **insert**, **count**, **distinct**, **group by**, **order by**, **sort by**, and **join**, are used to trigger MapReduce jobs. The following uses the procedure for assigning a role the permissions to run the **count** statements in the **thc** table as an example.

Step 1 Create a role on FusionInsight Manager.

Step 2 In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Yarn** > **Scheduler Queue** > **root**.

Step 3 In the **Permission** column of the **default** queue, select **Submit** and click **OK**.

Step 4 In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges** > **default**. Select **SELECT** for table **thc**, and click **OK**.

----End

Hive over HBase Authorization in MRS Earlier than 3.x

After the permissions are assigned, you can use HQL statements that are similar to SQL statements to access HBase tables from Hive. The following uses the procedure for assigning a user the rights to query HBase tables as an example.

Step 1 On the role management page of MRS Manager, create an HBase role, for example, **hive_hbase_create**, and grant the permission to create HBase tables.

In the **Permission** table, choose **HBase** > **HBase Scope** > **global**, select **create** of the namespace **default**, and click **OK**.

Step 2 On MRS Manager, create a human-machine user, for example, **hbase_creates_user**, add the user to the **hive** group, and bind the **hive_hbase_create** role to the user so that the user can create Hive and HBase tables.

Step 3 Log in to the node where the client is installed.

Step 4 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 5 Run the following command to authenticate the user:

kinit hbase_creates_user

Step 6 Run the following command to go to the shell environment of the Hive client:

beeline

Step 7 Run the following command to create a table in Hive and HBase, for example, the **thh** table.

CREATE TABLE thh(id int, name string, country string) STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES("hbase.columns.mapping" = "cf1:id,cf1:name,:key") TBLPROPERTIES ("hbase.table.name" = "thh");

The created Hive table and the HBase table are stored in the Hive database **default** and the HBase namespace **default**, respectively.

Step 8 On the role management page of MRS Manager, create a role, for example, **hive_hbase_select**, and assign the role the permission to query the Hive table **thh** and the HBase table **thh**.

1. In the **Permission** table, choose **HBase** > **HBase Scope** > **global** > **default**, select **Read** for the **thh** table, and click **OK** to grant the HBase role the permission to query the table.
2. Edit a role. In the **Permission** table, choose **HBase** > **HBase Scope** > **global** > **hbase**. Select **Execute** for **hbase:meta**, and click **OK**.
3. Edit a role. In the **Permission** table, choose **Hive** > **Hive Read Write Privileges** > **default**, select **Select** for **thh**, and click **OK**.

Step 9 On MRS Manager, create a human-machine user, for example, **hbase_select_user**, add the user to the **hive** group, and bind the **hive_hbase_select** role to the user so that the user can query Hive and HBase tables.

Step 10 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 11 Run the following command to authenticate users:

kinit hbase_select_user

Step 12 Run the following command to go to the shell environment of the Hive client:

beeline

Step 13 Run the following command to use an HQL statement to query HBase table data:

select * from thh;

----End

Hive over HBase Authorization in MRS 3.*x* or Later

After the permissions are assigned, you can use HQL statements that are similar to SQL statements to access HBase tables from Hive. The following uses the procedure for assigning a user the rights to query HBase tables as an example.

Step 1 On the role management page of FusionInsight Manager, create an HBase role, for example, **hive_hbase_create**, and grant the permission to create HBase tables.

In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global**. Select **Create** of the namespace **default**, and click **OK**.

Step 2 On FusionInsight Manager, create a human-machine user, for example, **hbase_creates_user**, add the user to the **hive** group, and bind the **hive_hbase_create** role to the user so that the user can create Hive and HBase tables.

Step 3 If the current component uses Ranger for permission control, grant the create permission for **hive_hbase_create** or **hbase_creates_user**. For details, see **Adding a Ranger Access Permission Policy for Hive**.

Step 4 Log in to the node where the client is installed as the client installation user.

Step 5 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 6 Run the following command to authenticate the user:

kinit hbase_creates_user

Step 7 Run the following command to go to the shell environment of the Hive client:

beeline

Step 8 Run the following command to create a table in Hive and HBase, for example, the **thh** table.

CREATE TABLE thh(id int, name string, country string) STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES("hbase.columns.mapping" = "cf1:id,cf1:name,:key") TBLPROPERTIES ("hbase.table.name" = "thh");

The created Hive table and the HBase table are stored in the Hive database **default** and the HBase namespace **default**, respectively.

Step 9 On the role management page of FusionInsight Manager, create a role, for example, **hive_hbase_select**, and assign the role the permission to query the Hive table **thh** and the HBase table **thh**.

1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global** > **default**. Select **read** of the **thh** table, and click **OK** to grant the table query permission to the HBase role.
2. Edit the role. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global** > **hbase**, select **Execute** for **hbase:meta**, and click **OK**.
3. Edit the role. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges** > **default**. Select **SELECT** for the **thh** table, and click **OK**.

Step 10 On FusionInsight Manager, create a human-machine user, for example, **hbase_select_user**, add the user to the **hive** group, and bind the **hive_hbase_select** role to the user so that the user can query Hive and HBase tables.

Step 11 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 12 Run the following command to authenticate users:

kinit hbase_select_user

Step 13 Run the following command to go to the shell environment of the Hive client:

beeline

Step 14 Run the following command to use an HQL statement to query HBase table data:

select * from thh;

----End

12.10.5 Using a Hive Client

Scenario

This section guides users to use a Hive client in an O&M or service scenario.

Prerequisites

- The client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login.

Using the Hive Client (Versions Earlier Than MRS 3.x)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Log in to the Hive client based on the cluster authentication mode.

- In security mode, run the following command to complete user authentication and log in to the Hive client:

  kinit *Component service user*

  beeline

- In common mode, run the following command to log in to the Hive client. If no component service user is specified, the current OS user is used to log in to the Hive client.

  beeline -n *component service user*

  NOTE

  After a beeline connection is established, you can compile and submit HQL statements to execute related tasks. To run the Catalog client command, you need to run the **!q** command first to exit the beeline environment.

Step 5 Run the following command to execute the HCatalog client command:

hcat -e *"cmd"*

cmd must be a Hive DDL statement, for example, **hcat -e "show tables"**.

NOTE

- To use the HCatalog client, choose **More** > **Download Client** on the service page to download the clients of all services. This restriction does not apply to the beeline client.
- Due to permission model incompatibility, tables created using the HCatalog client cannot be accessed on the HiveServer client. However, the tables can be accessed on the WebHCat client.
- If you use the HCatalog client in Normal mode, the system performs DDL commands using the current user who has logged in to the operating system.
- Exit the beeline client by running the **!q** command instead of by pressing **Ctrl + c**. Otherwise, the temporary files generated by the connection cannot be deleted and a large number of junk files will be generated as a result.
- If multiple statements need to be entered during the use of beeline clients, separate the statements from each other using semicolons (**;**) and set the value of **entireLineAsCommand** to **false**.

  Setting method: If beeline has not been started, run the **beeline --entireLineAsCommand=false** command. If the beeline has been started, run the **!set entireLineAsCommand false** command.

  After the setting, if a statement contains semicolons (**;**) that do not indicate the end of the statement, escape characters must be added, for example, **select concat_ws('\;', collect_set(col1)) from tbl**.

----End

Using the Hive Client (MRS 3.x or Later)

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 MRS 3.*X* supports multiple Hive instances. If you use the client to connect to a specific Hive instance in a scenario when multiple Hive instances are installed, run the following command to load the environment variables of the instance. Otherwise, skip this step. For example, load the environment variables of the Hive2 instance.

source Hive2/component_env

Step 5 Log in to the Hive client based on the cluster authentication mode.

- In security mode, run the following command to complete user authentication and log in to the Hive client:

  kinit *Component service user*

  beeline

- In common mode, run the following command to log in to the Hive client. If no component service user is specified, the current OS user is used to log in to the Hive client.

  beeline -n *component service user*

Step 6 Run the following command to execute the HCatalog client command:

hcat -e *"cmd"*

cmd must be a Hive DDL statement, for example, **hcat -e "show tables"**.

NOTE

- To use the HCatalog client, choose **More** > **Download Client** on the service page to download the clients of all services. This restriction does not apply to the beeline client.
- Due to permission model incompatibility, tables created using the HCatalog client cannot be accessed on the HiveServer client. However, the tables can be accessed on the WebHCat client.
- If you use the HCatalog client in Normal mode, the system performs DDL commands using the current user who has logged in to the operating system.
- Exit the beeline client by running the **!q** command instead of by pressing **Ctrl + C**. Otherwise, the temporary files generated by the connection cannot be deleted and a large number of junk files will be generated as a result.
- If multiple statements need to be entered during the use of beeline clients, separate the statements from each other using semicolons (**;**) and set the value of **entireLineAsCommand** to **false**.

  Setting method: If beeline has not been started, run the **beeline --entireLineAsCommand=false** command. If the beeline has been started, run the **!set entireLineAsCommand false** command.

  After the setting, if a statement contains semicolons (**;**) that do not indicate the end of the statement, escape characters must be added, for example, **select concat_ws('\;', collect_set(col1)) from tbl**.

----End

Common Hive Client Commands

The following table lists common Hive Beeline commands.

For more commands, see **https://cwiki.apache.org/confluence/display/Hive/HiveServer2+Clients#HiveServer2Clients-BeelineCommands**.

Table 12-235 Common Hive Beeline commands

Command	Description
set <key>=<value>	Sets the value of a specific configuration variable (key). **NOTE** If the variable name is incorrectly spelled, the Beeline does not display an error.
set	Prints the list of configuration variables overwritten by users or Hive.
set -v	Prints all configuration variables of Hadoop and Hive.
add FILE[S] <filepath> <filepath>*add JAR[S] <filepath> <filepath>*add ARCHIVE[S] <filepath> <filepath>*	Adds one or more files, JAR files, or ARCHIVE files to the resource list of the distributed cache.

Command	Description
add FILE[S] <ivyurl> <ivyurl>* add JAR[S] <ivyurl> <ivyurl>* add ARCHIVE[S] <ivyurl> <ivyurl>*	Adds one or more files, JAR files, or ARCHIVE files to the resource list of the distributed cache using the lvy URL in the **ivy://goup:module:version?query_string** format.
list FILE[S]list JAR[S]list ARCHIVE[S]	Lists the resources that have been added to the distributed cache.
list FILE[S] <filepath>*list JAR[S] <filepath>*list ARCHIVE[S] <filepath>*	Checks whether given resources have been added to the distributed cache.
delete FILE[S] <filepath>*delete JAR[S] <filepath>*delete ARCHIVE[S] <filepath>*	Deletes resources from the distributed cache.
delete FILE[S] <ivyurl> <ivyurl>* delete JAR[S] <ivyurl> <ivyurl>* delete ARCHIVE[S] <ivyurl> <ivyurl>*	Delete the resource added using **<ivyurl>** from the distributed cache.
reload	Enable HiveServer2 to discover the change of the JAR file **hive.reloadable.aux.jars.path** in the specified path. (You do not need to restart HiveServer2.) Change actions include adding, deleting, or updating JAR files.
dfs <dfs command>	Runs the **dfs** command.
<query string>	Executes the Hive query and prints the result to the standard output.

12.10.6 Using HDFS Colocation to Store Hive Tables

Scenario

HDFS Colocation is the data location control function provided by HDFS. The HDFS Colocation API stores associated data or data on which associated operations are performed on the same storage node. Hive supports the HDFS Colocation function. When Hive tables are created, after the locator information is set for table files, data files of related tables are stored on the same storage node when data is inserted into tables using the insert statement (other data import modes are not supported). This ensures convenient and efficient data computing among associated tables. The supported table formats are only TextFile and RCFile.

NOTE

This section applies to MRS 3.*x* or later.

Procedure

Step 1 Log in to the node where the client is installed as a client installation user.

Step 2 Run the following command to switch to the client installation directory, for example, **opt/client**:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user:

kinit *MRS username*

Step 5 Create the *groupid* through the HDFS API.

hdfs colocationadmin -createGroup -groupId *<groupid>* -locatorIds *<locatorid1>*,*<locatorid2>*,*<locatorid3>*

NOTE

In the preceding command, *<groupid>* indicates the name of the created group. The group created in this example contains three locators. You can define the number of locators as required.

For details about group ID creation and HDFS Colocation, see HDFS description.

Step 6 Run the following command to log in to the Hive client:

beeline

Step 7 Enable Hive to use colocation.

Assume that **table_name1** and **table_name2** are associated with each other. Run the following statements to create them:

CREATE TABLE *<[db_name.]table_name1>[(col_name data_type , ...)]* [ROW FORMAT *<row_format>*] [STORED AS *<file_format>*] TBLPROPERTIES("groupId"=" *<group>* ","locatorId"="*<locator1>*");

CREATE TABLE *<[db_name.]table_name2> [(col_name data_type , ...)]* [ROW FORMAT *<row_format>*] [STORED AS *<file_format>*] TBLPROPERTIES("groupId"=" *<group>* ","locatorId"="*<locator1>*");

After data is inserted into **table_name1** and **table_name2** using the insert statement, data files of **table_name1** and **table_name2** are distributed to the same storage position in the HDFS, facilitating associated operations among the two tables.

----End

12.10.7 Using the Hive Column Encryption Function

Scenario

Hive supports encryption of one or multiple columns in a table. When creating a Hive table, you can specify the column to be encrypted and encryption algorithm. When data is inserted into the table using the insert statement, the related columns are encrypted. Column encryption can be performed in HDFS tables of only the TextFile and SequenceFile file formats. The Hive column encryption does not support views and the Hive over HBase scenario.

Hive supports two column encryption algorithms, which can be specified during table creation:

- AES (the encryption class is org.apache.hadoop.hive.serde2.AESRewriter)
- SMS4 (the encryption class is org.apache.hadoop.hive.serde2.SMS4Rewriter)

NOTE

- In national cryptographic cluster scenarios, Hive column encryption supports only table creation using the SMS4 algorithm.
- When you import data from a common Hive table into a Hive column encryption table, you are advised to delete the original data from the common Hive table as long as doing this does not affect other services. Retaining an unencrypted table poses security risks.

Procedure

Step 1 Specify the column to be encrypted and encryption algorithm when creating a table.

create table*<[db_name.]table_name> (<col_name1> <data_type> ,<col_name2> <data_type>,<col_name3> <data_type>,<col_name4> <data_type>)* ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe' WITH SERDEPROPERTIES ('column.encode.columns'='*<col_name2>,<col_name3>*', 'column.encode.classname'='org.apache.hadoop.hive.serde2.AESRewriter')STORED AS TEXTFILE;

Alternatively, use the following statement:

create table *<[db_name.]table_name> (<col_name1> <data_type> ,<col_name2> <data_type>,<col_name3> <data_type>,<col_name4> <data_type>)* ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe' WITH SERDEPROPERTIES ('column.encode.indices'='1,2', 'column.encode.classname'='org.apache.hadoop.hive.serde2.SMS4Rewriter') STORED AS TEXTFILE;

NOTE

- The numbers used to specify encryption columns start from 0. 0 indicates column 1, 1 indicates column 2, and so on.
- When creating a table with encrypted columns, ensure that the directory where the table resides is empty.

Step 2 Insert data into the table using the insert statement.

Assume that the test table exists and contains data.

insert into table *<table_name>* select *<col_list>* from test;

----End

12.10.8 Customizing Row Separators

Scenario

In most cases, a carriage return character is used as the row delimiter in Hive tables stored in text files, that is, the carriage return character is used as the terminator of a row during queries. However, some data files are delimited by special characters, and not a carriage return character.

MRS Hive allows you to use different characters or character combinations to delimit rows of Hive text data. When creating a table, set **inputformat** to **SpecifiedDelimiterInputFormat**, and set the following parameter before search each time. Then the table data is queried by the specified delimiter.

set hive.textinput.record.delimiter='';

NOTE

- The Hue component of the current version does not support the configuration of multiple separators when files are imported to a Hive table.
- This section applies to MRS 3.*x* or later.

Procedure

Step 1 Specify **inputFormat** and **outputFormat** when creating a table.

CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS] *[db_name.]table_name* **[(***col_name data_type* **[COMMENT** *col_comment***], ...)] [ROW FORMAT** *row_format***] STORED AS inputformat 'org.apache.hadoop.hive.contrib.fileformat.SpecifiedDelimiterInputFormat' outputformat 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'**

Step 2 Specify the delimiter before search.

set hive.textinput.record.delimiter='!@!'

Hive will use '!@!' as the row delimiter.

----End

12.10.9 Configuring Hive on HBase in Across Clusters with Mutual Trust Enabled

For mutually trusted Hive and HBase clusters with Kerberos authentication enabled, you can access the HBase cluster and synchronize its key configurations to HiveServer of the Hive cluster.

Prerequisites

The mutual trust relationship has been configured between the two security clusters with Kerberos authentication enabled.

Procedure for Configuring Hive on HBase Across Clusters

Step 1 Download the HBase configuration file and decompress it.

1. Log in to FusionInsight Manager of the target HBase cluster and choose **Cluster** > **Services** > **HBase**.
2. Choose **More** > **Download Client**.
3. Download the HBase configuration file and choose **Configuration Files only** for **Select Client Type**.

Step 2 Log in to FusionInsight Manager of the source Hive cluster.

Step 3 Choose **Cluster** > **Services** > **Hive** and click the **Configurations** tab and then **All Configurations**. On the displayed page, add the following parameters to the **hive-site.xml** configuration file of the HiveServer role.

Search for the following parameters in the **hbase-site.xml** configuration file of the downloaded HBase client and add them to HiveServer:

- hbase.security.authentication
- hbase.security.authorization
- hbase.zookeeper.property.clientPort
- hbase.zookeeper.quorum (The domain name needs to be converted into an IP address.)
- hbase.regionserver.kerberos.principal
- hbase.master.kerberos.principal

Step 4 Save the configurations and restart Hive.

----End

12.10.10 Deleting Single-Row Records from Hive on HBase

Scenario

Due to the limitations of underlying storage systems, Hive does not support the ability to delete a single piece of table data. In Hive on HBase, MRS Hive supports the ability to delete a single piece of HBase table data. Using a specific syntax, Hive can delete one or more pieces of data from an HBase table.

Table 12-236 Permissions required for deleting single-row records from the Hive on HBase table

Cluster Authentication Mode	Required Permission
Security mode	SELECT, INSERT, and DELETE
Common mode	None

Procedure

Step 1 To delete some data from an HBase table, run the following HQL statement:

remove table *<table_name>* where *<expression>*;

In the preceding information, *<expression>* specifies the filter condition of the data to be deleted. *<table_name>* indicates the Hive on HBase table from which data is to be deleted.

----End

12.10.11 Configuring HTTPS/HTTP-based REST APIs

Scenario

WebHCat provides external REST APIs for Hive. By default, the open-source community version uses the HTTP protocol.

MRS Hive supports the HTTPS protocol that is more secure, and enables switchover between the HTTP protocol and the HTTPS protocol.

NOTE

The security mode supports HTTPS and HTTP, and the common mode supports only HTTP.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Modify the Hive configuration.

- For versions earlier than MRS 3.x: Enter the parameter name in the search box, search for **templeton.protocol.type**, change the parameter value to **HTTPS** or **HTTP**, and restart the Hive service to use the corresponding protocol.
- For MRS 3.*x* or earlier: Choose **WebHCat** > **Security**. On the page that is displayed, select **HTTPS** or **HTTP**. After the modification, restart the Hive service to use the corresponding protocol.

----End

12.10.12 Enabling or Disabling the Transform Function

Scenario

The Transform function is not allowed by Hive of the open source version.

MRS Hive supports the configuration of the Transform function. The function is disabled by default, which is the same as that of the open-source community version.

Users can modify configurations of the Transform function to enable the function. However, security risks exist when the Transform function is enabled.

NOTE

The Transform function can be disabled only in security mode.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Enter the parameter name in the search box, search for **hive.security.transform.disallow**, change the parameter value to **true** or **false**, and restart all HiveServer instances.

NOTE

- If this parameter is set to **true**, the Transform function is disabled, which is the same as that in the open-source community version.
- If this parameter is set to **false**, the Transform function is enabled, which poses security risks.

----End

12.10.13 Access Control of a Dynamic Table View on Hive

Scenario

This section describes how to create a view on Hive when MRS is configured in security mode, authorize access permissions to different users, and specify that different users access different data.

In the view, Hive can obtain the built-in function **current_user()** of the users who submit tasks on the client and filter the users. This way, authorized users can only access specific data in the view.

NOTE

In normal mode, the **current_user()** function cannot distinguish users who submit tasks on the client. Therefore, the access control function takes effect only for Hive in security mode.

If the **current_user()** function is used in the actual service logic, the possible risks must be fully evaluated during the conversion between the security mode and normal mode.

Operation Example

- If the current_user function is not used, different views need to be created for different users to access different data.
  - Authorize the view **v1** permission to user **hiveuser1**. The user **hiveuser1** can access data with **type** set to **hiveuser1** in **table1**.

    create view v1 as select * from table1 where type='hiveuser1'
  - Authorize the view **v2** permission to user **hiveuser2**. The user **hiveuser2** can access data with **type** set to **hiveuser2** in **table1**.

    create view v2 as select * from table1 where type='hiveuser2'
- If the current_user function is used, only one view needs to be created.

  Authorize the view **v** permission to users **hiveuser1** and **hiveuser2**. When user **hiveuser1** queries view **v**, the current_user() function is automatically converted to **hiveuser1**. When user **hiveuser2** queries view **v**, the **current_user()** function is automatically converted to **hiveuser2**.

  create view v as select * from table1 where type=current_user()

12.10.14 Specifying Whether the ADMIN Permissions Is Required for Creating Temporary Functions

Scenario

You must have **ADMIN** permission when creating temporary functions on Hive of the open source community version.

MRS Hive supports the configuration of the function for creating temporary functions with **ADMIN** permission. The function is disabled by default, which is the same as that of the open-source community version.

You can modify configurations of this function. After the function is enabled, you can create temporary functions without **ADMIN** permission. If this parameter is set to **false**, security risks exist.

NOTE

The security mode supports the configuration of whether the ADMIN permission is required for creating temporary functions, but the common mode does not support this function.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Enter the parameter name in the search box, search for **hive.security.temporary.function.need.admin**, change the parameter value to **true** or **false**, and restart all HiveServer instances.

NOTE

- If this parameter is set to **true**, the ADMIN permission is required for creating temporary functions, which is the same as that in the open source community.
- If this parameter is set to **false**, the ADMIN permission is not required for creating temporary functions.

----End

12.10.15 Using Hive to Read Data in a Relational Database

Scenario

Hive allows users to create external tables to associate with other relational databases. External tables read data from associated relational databases and support Join operations with other tables in Hive.

Currently, the following relational databases can use Hive to read data:

- DB2
- Oracle

NOTE

This section applies to MRS 3.*x* or later.

Prerequisites

The Hive client has been installed.

Procedure

Step 1 Log in to the node where the Hive client is installed as the Hive client installation user .

Step 2 Run the following command to go to the client installation directory:

cd *Client installation directory*

For example, if the client installation directory is **/opt/client**, run the following command:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Check whether the cluster authentication mode is Security.

- If yes, run the following command to authenticate the user:

  kinit *Hive service user*

- If no, go to **Step 5**.

Step 5 Run the following command to upload the driver JAR package of the relational database to be associated to an HDFS directory.

hdfs dfs -put *directory where the JAR package is located HDFS directory to which the JAR is uploaded*

For example, to upload the Oracle driver JAR package in **/opt** to the **/tmp** directory in HDFS, run the following command:

hdfs dfs -put /opt/ojdbc6.jar /tmp

Step 6 Create an external table on the Hive client to associate with the relational database, as shown in the following example.

NOTE

If the security mode is used, the user who creates the table must have the **ADMIN** permission. The **ADD JAR** path is subject to the actual path.

```
-- Example of associating with an Oracle Linux 6 database
-- In security mode, set the admin permission.
set role admin;
-- Upload the driver JAR package of the relational database to be associated. The driver JAR packages vary according to databases.
ADD JAR hdfs:///tmp/ojdbc6.jar;

CREATE EXTERNAL TABLE ora_test
-- The Hive table must have one more column than the database return result. This column is used for paging query.
(id STRING,rownum string)
STORED BY 'com.qubitproducts.hive.storage.jdbc.JdbcStorageHandler'
TBLPROPERTIES (
-- Relational database table type
"qubit.sql.database.type" = "ORACLE",
-- Connect to the URL of the relational database through JDBC. (The URL formats vary according to databases.)
"qubit.sql.jdbc.url" = "jdbc:oracle:thin:@//10.163.0.1:1521/mydb",
-- Relational database driver class type
"qubit.sql.jdbc.driver" = "oracle.jdbc.OracleDriver",
-- SQL statement queried in the relational database. The result is returned to the Hive table.
"qubit.sql.query" = "select name from aaa",
-- (Optional) Match the Hive table columns to the relational database table columns.
"qubit.sql.column.mapping" = "id=name",
-- Relational database user
"qubit.sql.dbcp.username" = "test",
-- Relational database password
"qubit.sql.dbcp.password" = "xxx");
```

----End

12.10.16 Supporting Traditional Relational Database Syntax in Hive

Overview

Hive supports the following types of traditional relational database syntax:

- Grouping
- EXCEPT and INTERSECT

Grouping

Syntax description:

- Grouping takes effect only when the Group by statement contains ROLLUP or CUBE.
- The result set generated by CUBE contains all the combinations of values in the selected columns.
- The result set generated by ROLLUP contains the combinations of a certain layer structure in the selected columns.
- Grouping: If a row is added by using the CUBE or ROLLUP operator, the output value of the added row is 1. If the row is not added by using the CUBE or ROLLUP operator, the output value of the added row is 0.

For example, the **table_test** table exists in Hive and the table structure is as follows:

```
+----------------+-------------------+--+
| table_test.id  | table_test.value  |
+----------------+-------------------+--+
| 1              | 10                |
| 1              | 15                |
| 2              | 20                |
| 2              | 5                 |
| 2              | 13                |
+----------------+-------------------+--+
```

Run the following statement:

select id,grouping(id),sum(value) from table_test group by id with rollup;

The result is as follows:

```
+-------+-----------------+------+--+
|  id   | groupingresult  | sum  |
+-------+-----------------+------+--+
| 1     | 0               | 25   |
| NULL  | 1               | 63   |
| 2     | 0               | 38   |
+-------+-----------------+------+--+
```

EXCEPT and INTERSECT

Syntax description:

- EXCEPT returns the difference of two result sets (that is, non-duplicated values return only one query).

- INTERSECT returns the intersection of two result sets (that is, non-duplicated values return by both queries).

For example, two tables **test_table1** and **test_table2** exist in Hive.

The table structure of **test_table1** is as follows:

```
+-----------------+--+
| test_table1.id  |
+-----------------+--+
| 1               |
| 2               |
| 3               |
| 4               |
+-----------------+--+
```

The table structure of **test_table2** is as follows:

```
+-----------------+--+
| test_table2.id  |
+-----------------+--+
| 2               |
| 3               |
| 4               |
| 5               |
+-----------------+--+
```

- Run the following EXCEPT statement:

  select id from test_table1 except select id from test_table2;

  The result is as follows:

  ```
  +--------------+--+
  | _alias_0.id  |
  +--------------+--+
  | 1            |
  +--------------+--+
  ```

- Run the following INTERSECT statement:

  select id from test_table1 intersect select id from test_table2;

  The result is as follows:

  ```
  +--------------+--+
  | _alias_0.id  |
  +--------------+--+
  | 2            |
  | 3            |
  | 4            |
  +--------------+--+
  ```

12.10.17 Creating User-Defined Hive Functions

When built-in functions of Hive cannot meet requirements, you can compile user-defined functions (UDFs) and use them for query.

According to implementation methods, UDFs are classified as follows:

- Common UDFs: used to perform operations on a single data row and export a single data row.
- User-defined aggregating functions (UDAFs): used to input multiple data rows and export a single data row.
- User-defined table-generating functions (UDTFs): used to perform operations on a single data row and export multiple data rows.

According to use methods, UDFs are classified as follows:

- Temporary functions: used only in the current session and must be recreated after a session restarts.
- Permanent functions: used in multiple sessions. You do not need to create them every time a session restarts.

NOTE

You need to properly control the memory and thread usage of variables in UDFs. Improper control may cause memory overflow or high CPU usage.

The following uses AddDoublesUDF as an example to describe how to compile and use UDFs.

Function

AddDoublesUDF is used to add two or more floating point numbers. In this example, you can learn how to write and use UDFs.

NOTE

- A common UDF must be inherited from **org.apache.hadoop.hive.ql.exec.UDF**.
- A common UDF must implement at least one **evaluate()**. The evaluate function supports overloading.
- To develop a customized function, you need to add the **hive-exec-3.1.0.jar** dependency package to the project. The package can be obtained from the Hive installation directory.

Sample Code

The following is a UDF code example:

xxx indicates the name of the organization that develops the program.

```
package com.xxx.bigdata.hive.example.udf;
import org.apache.hadoop.hive.ql.exec.UDF;

public class AddDoublesUDF extends UDF {
 public Double evaluate(Double... a) {
    Double total = 0.0;
    // Processing logic
    for (int i = 0; i < a.length; i++)
      if (a[i] != null)
        total += a[i];
    return total;
  }
}
```

How to Use

Step 1 Packing programs as **AddDoublesUDF.jar** on the client node, and upload the package to a specified directory in HDFS, for example, **/user/hive_examples_jars**.

Both the user who creates the function and the user who uses the function must have the read permission on the file.

The following are example statements:

hdfs dfs -put ./hive_examples_jars /user/hive_examples_jars

hdfs dfs -chmod 777 /user/hive_examples_jars

Step 2 Check the cluster authentication mode.

- In security mode, log in to the beeline client as a user with the Hive management permission and run the following commands:

  kinit *Hive service user*

  beeline

  set role admin;

- In common mode, run the following command:

  beeline -n *Hive service user*

Step 3 Define the function in HiveServer. Run the following SQL statement to create a permanent function:

CREATE FUNCTION *addDoubles* **AS 'com.***xxx***.bigdata.hive.example.udf.AddDoublesUDF' using jar 'hdfs://hacluster/user/hive_examples_jars/AddDoublesUDF.jar';**

addDoubles indicates the function alias that is used for SELECT query. *xxx* indicates the name of the organization that develops the program.

Run the following statement to create a temporary function:

CREATE TEMPORARY FUNCTION addDoubles AS 'com.*xxx***.bigdata.hive.example.udf.AddDoublesUDF' using jar 'hdfs://hacluster/user/hive_examples_jars/AddDoublesUDF.jar';**

- *addDoubles* indicates the function alias that is used for SELECT query.
- **TEMPORARY** indicates that the function is used only in the current session with the HiveServer.

Step 4 Run the following SQL statement to use the function on the HiveServer:

SELECT addDoubles(1,2,3);

NOTE

If an [Error 10011] error is displayed when you log in to the client again, run the **reload function;** command and then use this function.

Step 5 Run the following SQL statement to delete the function from the HiveServer:

DROP FUNCTION addDoubles;

----End

Extended Applications

None

12.10.18 Enhancing beeline Reliability

Scenario

- When the beeline client is disconnected due to network exceptions during the execution of a batch processing task, tasks submitted before beeline is disconnected can be properly executed in Hive. When you start the batch

processing task again, the submitted tasks are not executed and tasks that are not executed are executed in sequence.

- When the HiveServer service breaks down due to some reasons during the execution of a batch processing task, Hive enables that the tasks that have been successfully executed are not executed again when the same batch processing task is started again. The execution starts from the task that has not been executed from the time when HiveServer2 breaks down.

NOTE

This section applies to MRS 3.*x* or later.

Example

1. Beeline is reconnected after being disconnection.

   Example:

   beeline -e "${*SQL*}" --hivevar batchid=*xxxxx*

2. Beeline kills the running tasks.

   Example:

   beeline -e "" --hivevar batchid=*xxxxx* --hivevar kill=true

3. Log in to the beeline client and start the mechanism of reconnection after disconnection.

   Log in to the beeline client and run the **set hivevar:batchid=***xxxx* command.

NOTE

Instructions:

- *xxxx* indicates the batch ID of tasks submitted in the same batch using the beeline client. Batch IDs can be used to identify the task submission batch. If the batch ID is not contained when a task is submitted, this feature is not enabled. The value of *xxxx* is specified during task execution. In the following example, the value of *xxxx* is **012345678901**.

  beeline -f hdfs://hacluster/user/hive/table.sql --hivevar batchid=012345678901
- If the running SQL script depends on the data timeliness, you are advised not to enable the breakpoint reconnection mechanism. You can use a new batch ID to submit tasks. During reexecution of the scripts, some SQL statements have been executed and are not executed again. As a result, expired data is obtained.
- If some built-in time functions are used in the SQL script, it is recommended that you do not enable the breakpoint reconnection mechanism or the use of a new batch ID for each execution. The reason is the same as above.
- A SQL script contains one or more subtasks. If the logic for deleting and creating temporary tables exist in the SQL script, it is recommended that the logic for deleting temporary tables be placed at the end of the script. If the subtasks executed after the temporary table deletion task fail to be executed and the temporary table is used in the subtasks before the temporary table deletion task, when the SQL script is executed using the same batch ID for the next time, the compilation of the subtasks (excluding the task for creating the temporary table because the creation has been completed and is not executed again, and only compilation is allowed) executed before the temporary table deletion task fails because the temporary has been deleted. In this case, you are advised to use a new batch ID to execute the script.

Parameter description:

- **zk.cleanup.finished.job.interval**: indicates the interval for executing the cleanup task. The default interval is 60 seconds.
- **zk.cleanup.finished.job.outdated.threshold**: indicates the threshold of the node validity period. A node is generated for tasks in the same batch. The threshold is calculated from the end time of the execution of the current batch task. If the time exceeds 60 minutes, the node is deleted.
- **batch.job.max.retry.count**: indicates the maximum number of retry times of a batch task. If the number of retry times of a batch task exceeds the value of this parameter, the task execution record is deleted. The task will be executed from the first task when the task is started next time. The default value is **10**.
- **beeline.reconnect.zk.path**: indicates the root node for storing task execution progress. The default value for the Hive service is **/beeline**.

12.10.19 Viewing Table Structures Using the show create Statement as Users with the select Permission

Scenario

This function is applicable to Hive and Spark2x in MRS 3.x and later.

With this function enabled, if the select permission is granted to a user during Hive table creation, the user can run the **show create table** command to view the table structure.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.allow.show.create.table.in.select.nogrant**, and set **Value** to **true**. Restart all Hive instances after the modification.

Step 3 Determine whether to enable this function on the Spark/Spark2x client.

- If yes, download and install the Spark/Spark2x client again.
- If no, no further action is required.

----End

12.10.20 Writing a Directory into Hive with the Old Data Removed to the Recycle Bin

Scenario

This function applies to Hive.

After this function is enabled, run the following command to write a directory into Hive: **insert overwrite directory "***/path1***" ...**. After the operation is successfully performed, the old data is removed to the recycle bin, and the directory cannot be an existing database path in the Hive metastore.

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.overwrite.directory.move.trash**, and set **Value** to **true**. Restart all Hive instances after the modification.

----End

12.10.21 Inserting Data to a Directory That Does Not Exist

Scenario

This function applies to Hive.

With this function enabled, run the **insert overwrite directory** */path1/path2/path3*... command to write a subdirectory. The permission of the */path1/path2* directory is 700, and the owner is the current user. If the */path3* directory does not exist, it is automatically created and data is written successfully.

This function is supported when **hive.server2.enable.doAs** is set to **true** in earlier versions. This version supports the function when **hive.server2.enable.doAs** is set to **false**.

NOTE

The parameter adjustment of this function is the same as that of the custom parameters added in **Writing a Directory into Hive with the Old Data Removed to the Recycle Bin**.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.overwrite.directory.move.trash**, and set **Value** to **true**. Restart all Hive instances after the modification.

----End

12.10.22 Creating Databases and Creating Tables in the Default Database Only as the Hive Administrator

Scenario

This function is applicable to Hive and Spark2x for MRS 3.*x* or later, or Hive and Spark for versions earlier than MRS 3.x.

After this function is enabled, only the Hive administrator can create databases and tables in the default database. Other users can use the databases only after being authorized by the Hive administrator.

NOTE

- After this function is enabled, common users are not allowed to create a database or create a table in the default database. Based on the actual application scenario, determine whether to enable this function.
- Permissions of common users are restricted. In the scenario where common users have been used to perform operations, such as database creation, table script migration, and metadata recreation in an earlier version of database, the users can perform such operations on the database in the condition that this function is disabled temporarily after the database is migrated or after the cluster is upgraded.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.allow.only.admin.create**, and set **Value** to **true**. Restart all Hive instances after the modification.

Step 3 Determine whether to enable this function on the Spark/Spark2x client.

- If yes, go to **Step 4**.
- If no, no further action is required.

Step 4 Choose **SparkResource2x** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.allow.only.admin.create**, and set **Value** to **true**. Then, choose **JDBCServer2x** > **Customization** and repeat the preceding operations to add the customized parameter. Restart all Spark2x instances after the modification.

Step 5 Download and install the Spark/Spark2x client again.

----End

12.10.23 Disabling of Specifying the location Keyword When Creating an Internal Hive Table

Scenario

This function is applicable to Hive and Spark2x for MRS 3.*x* or later, or Hive and Spark for versions earlier than MRS 3.x.

After this function is enabled, the **location** keyword cannot be specified when a Hive internal table is created. Specifically, after a table is created, the table path following the location keyword is created in the default **\warehouse** directory and cannot be specified to another directory. If the location is specified when the internal table is created, the creation fails.

NOTE

After this function is enabled, the location keyword cannot be specified during the creation of a Hive internal table. The table creation statement is restricted. If a table that has been created in the database is not stored in the default directory **/warehouse**, the **location** keyword can still be specified when the database creation, table script migration, or metadata recreation operation is performed by disabling this function temporarily.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.internaltable.notallowlocation**, and set **Value** to **true**. Restart all Hive instances after the modification.

Step 3 Determine whether to enable this function on the Spark/Spark2x client.

- If yes, download and install the Spark/Spark2x client again.
- If no, no further action is required.

----End

12.10.24 Enabling the Function of Creating a Foreign Table in a Directory That Can Only Be Read

Scenario

This function is applicable to Hive and Spark2x for MRS 3.*x* or later, or Hive and Spark for versions earlier than MRS 3.x.

After this function is enabled, the user or user group that has the read and execute permissions on a directory can create foreign tables in the directory without checking whether the current user is the owner of the directory. In addition, the directory of a foreign table cannot be stored in the default directory **\warehouse**. In addition, do not change the permission of the directory during foreign table authorization.

NOTE

After this function is enabled, the function of the foreign table changes greatly. Based on the actual application scenario, determine whether to enable this function.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.restrict.create.grant.external.table**, and set **Value** to **true**.

Step 3 Choose **MetaStore(Role)** > **Customization**, add a customized parameter to the **hivemetastore-site.xml** parameter file, set **Name** to **hive.restrict.create.grant.external.table**, and set **Value** to **true**. Restart all Hive instances after the modification.

Step 4 Determine whether to enable this function on the Spark/Spark2x client.

- If yes, download and install the Spark/Spark2x client again.
- If no, no further action is required.

----End

12.10.25 Authorizing Over 32 Roles in Hive

Scenario

This function applies to Hive.

The number of OS user groups is limited, and the number of roles that can be created in Hive cannot exceed 32. After this function is enabled, more than 32 roles can be created in Hive.

NOTE

- After this function is enabled and the table or database is authorized, roles that have the same permission on the table or database will be combined using vertical bars (|). When the ACL permission is queried, the combined result is displayed, which is different from that before the function is enabled. This operation is irreversible. Determine whether to make adjustment based on the actual application scenario.
- MRS 3.*x* and later versions support Ranger. If the current component uses Ranger for permission control, you need to configure related policies based on Ranger for permission management. For details, see **Adding a Ranger Access Permission Policy for Hive**.
- After this function is enabled, a maximum of 512 roles (including **owner**) are supported by default. The number is controlled by the user-defined parameter **hive.supports.roles.max** of MetaStore. You can change the value based on the actual application scenario.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **MetaStore(Role)** > **Customization**, add a customized parameter to the **hivemetastore-site.xml** parameter file, set **Name** to **hive.supports.over.32.roles**, and set **Value** to **true**. Restart all Hive instances after the modification.

Step 3 Choose **HiveServer(Role)** > **Customization**, add a customized parameter to the **hive-site.xml** parameter file, set **Name** to **hive.supports.over.32.roles**, and set **Value** to **true**. Restart all Hive instances after the modification.

----End

12.10.26 Restricting the Maximum Number of Maps for Hive Tasks

Scenario

- This function applies to Hive.
- This function is used to limit the maximum number of maps for Hive tasks on the server to avoid performance deterioration caused by overload of the HiveSever service.

Procedure

Step 1 The Hive service configuration page is displayed.

- For versions earlier than MRS 3.x, click the cluster name. On the cluster details page that is displayed, choose **Components** > **Hive** > **Service Configuration**, and select **All** from the **Basic** drop-down list.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. And choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 2 Choose **MetaStore(Role)** > **Customization**, add a customized parameter to the **hivemetastore-site.xml** parameter file, set **Name** to **hive.mapreduce.per.task.max.splits**, and set the parameter to a large value. Restart all Hive instances after the modification.

----End

12.10.27 HiveServer Lease Isolation

Scenario

- This function applies to Hive.
- This function can be enabled to specify specific users to access HiveServer services on specific nodes, achieving HiveServer resource isolation.

NOTE

This section applies to MRS 3.*x* or later.

Procedure

This section describes how to set lease isolation for user **hiveuser** for existing HiveServer instances.

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **HiveServer**.

Step 3 In the HiveServer list, select the HiveServer for which lease isolation is configured and choose **HiveServer** > **Instance Configurations** > **All Configurations**.

Step 4 In the upper right corner of the **All Configurations** page, search for **hive.server2.zookeeper.namespace** and specify its value, for example, **hiveserver2_zk**.

Step 5 Click **Save**. In the dialog box that is displayed, click **OK**.

Step 6 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive**, choose **More** > **Restart Service**, and enter the password to restart the service.

Step 7 Run the **beeline -u** command to log in to the client and run the following command:

beeline -u "jdbc:hive2:// *10.5.159.13* **:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=** *hiveserver2_zk* **;sasl.qop=auth-conf;auth=KERBEROS;principal=hive/hadoop.** *<System domain name>* **@** *<System domain name>* **"**

In the command, **10.5.159.13** is replaced with the IP address of any ZooKeeper instance, which can be viewed through **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Instance**.

hiveserver2_zk following **zooKeeperNamespace=** is set to the value of **hive.server2.zookeeper.namespace** in **Step 4**.

As a result, only the HiveServer whose lease isolation is configured can be logged in.

NOTE

- After this function is enabled, you must run the preceding command during login to access the HiveServer for which lease isolation is configured. If you run the **beeline** command to log in to the client, only the HiveServer that is not isolated by the lease is accessed.
- You can log in to FusionInsight Manager, choose **System** > **Permission** > **Domain and Mutual Trust**, and view the value of **Local Domain**, which is the current system domain name. **hive/hadoop.**<*system domain name*> is the username. All letters in the system domain name contained in the username are lowercase letters.

----End

12.10.28 Hive Supporting Transactions

Scenario

Hive supports transactions at the table and partition levels. When the transaction mode is enabled, transaction tables can be incrementally updated, deleted, and read, implementing atomicity, isolation, consistency, and durability of operations on transaction tables.

NOTE

This section applies to MRS 3.*x* or later.

Introduction to Transaction Features

A transaction is a group of unitized operations. These operations are either executed together or not executed together. A transaction is an inseparable unit of work. The four basic elements of a transaction are usually called ACID features, which are as follows:

- Atomicity: A transaction is an inseparable unit of work. All operations in a transaction occur or do not occur together.
- Consistency: The database integrity constraints are not damaged before and after a transaction starts.
- Isolation: When multiple transactions are concurrently accessed, the transactions are isolated from each other. A transaction does not affect the running of other transactions. The impacts between transactions are as follows: dirty read, non-repeatable read, phantom read, and lost update.
- Durability: After a transaction is complete, changes made by the transaction lock to the database are permanently stored in the database.

Characteristics of transaction execution:

- A statement can be written to multiple partitions or tables. If the operation fails, the user cannot see partial write or insert. Even if data is frequently changed, operations can still be quickly performed.
- Hive can automatically compress ACID transaction files without affecting concurrent queries. When querying many small partition files, automatic compression can improve query performance and metadata occupation.
- Read semantics include snapshot isolation. When the read operation starts, the Hive data warehouse is logically locked. The read operation is not affected by any changes that occur during the operation.

Lock Mechanism

Transactions implement the ACID feature through the following two aspects:

- Write-ahead logging ensures atomicity and durability.
- Locking ensures isolation.

Operation	Type of Held Locks
Insert overwrite	If **hive.txn.xlock.iow** is set to **true**, the exclusive lock is held. If **hive.txn.xlock.iow** is set to **false**, the semi-shared lock is held.
Insert	Shared lock. When performing this operation, you can perform read and write operations on the current table or partition.
Update/delete	Semi-shared lock. When this operation is performed, an operation of holding a shared lock can be performed, but an operation of holding an exclusive lock or a semi-shared lock cannot be performed.

Operation	Type of Held Locks
Drop	Exclusive lock. You cannot perform any other operations on the current table or partition when performing this operation.

NOTE

If a conflict caused by the lock mechanism exists in the write operation, the operation that preferentially holds the lock succeeds, and other operations fail.

Procedure

Starting a Transaction

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations** > **MetaStore(Role)** > **Transaction**.

Step 2 Set **metastore.compactor.initiator.on** to **true**.

Step 3 Set **metastore.compactor.worker.threads** to a positive integer.

NOTE

metastore.compactor.worker.threads: Specifies the number of working threads for running the compression program on MetaStore. Set this parameter based on the actual requirements. If the value is too small, the transaction compression task is executed slowly. If the value is too large, the MetaStore execution performance deteriorates.

Step 4 Log in to the Hive client and run the following command to enable the following parameters. For details, see **Using a Hive Client**.

set hive.support.concurrency=true;

set hive.exec.dynamic.partition.mode=nonstrict;

set hive.txn.manager=org.apache.hadoop.hive.ql.lockmgr.DbTxnManager;

Create a transaction table.

Step 5 Run the following command to create a transaction table:

CREATE TABLE [IF NOT EXISTS] *[db_name.]table_name* **(***col_name data_type [COMMENT col_comment], ...***) [ROW FORMAT row_format] STORED AS orc ..... TBLPROPERTIES ('transactional'='true'[,'groupId'='group1' ...]);**

For example:

CREATE TABLE acidTbl (a int, b int) STORED AS ORC TBLPROPERTIES ('transactional'='true');

NOTE

- Currently, the transactions support only the ORC format.
- External tables are not supported.
- Sorted tables are not supported.
- To create a transaction table, you must add the table attribute **transactional'='true'**.
- The transaction table can be read and written only in transaction mode.

Use the transaction table.

Step 6 Run commands to use the transaction table. The following uses the **acidTbl** table as an example:

- Insert data into an existing transaction table:

  INSERT INTO acidTbl VALUES(1,1);

- Update an existing transaction table:

  UPDATE acidTbl SET b = 10 where a = 1;

  The content of **acidTbl** is changed to:

  ```
  +------------+------------+
  | acidtbl.a  | acidtbl.b  |
  +------------+------------+
  | 1          | 10         |
  +------------+------------+
  1 row selected (0.775 seconds)
  ```

- Merge the old and new transaction tables:

  The **acidTbl_update** table contains the following data:

  ```
  +--------------------+--------------------+
  | acidtbl_update.a   | acidtbl_update.b   |
  +--------------------+--------------------+
  | 1                  | 20                 |
  | 2                  | 10                 |
  +--------------------+--------------------+
  2 rows selected (0.537 seconds)
  ```

  MERGE INTO acidTbl AS a

  USING acidTbl_update AS b ON a.a = b.a

  WHEN MATCHED THEN UPDATE SET b = b. b

  WHEN NOT MATCHED THEN INSERT VALUES (b.a, b.b);

  The content of **acidTbl** is changed to:

  ```
  +------------+------------+
  | acidtbl.a  | acidtbl.b  |
  +------------+------------+
  | 1          | 20         |
  | 2          | 10         |
  +------------+------------+
  2 rows selected (0.666 seconds)
  ```

  NOTE

  If "Error evaluating cardinality_violation" is displayed when you run the **merge** command, check whether duplicate connection keys exist or run the **set hive.merge.cardinality.check=false** command to avoid this exception.

- Delete records from the transaction table.

  DELETE FROM acidTbl where a = 2;

  ```
  +------------+------------+
  | acidtbl.a  | acidtbl.b  |
  +------------+------------+
  | 1          | 20         |
  +------------+------------+
  1 row selected (1.253 seconds)
  ```

Checking the Transaction Execution Status

Step 7 Run the following command to check the transaction execution status:

- Check the lock:

  show locks;

- Check the compression task:

  show compactions;

- Check the task execution status:

  show transactions;

- Interrupt a transaction:

  abort transactions *TransactionId*;

  TransactionId is the value in the **Transaction ID** column in the command output of **Check the task execution status**.

----End

Configuring the Compression Function

HDFS does not support in-place file changing. For the new content, HDFS does not provide read consistency either. To provide these features on HDFS, we follow the standard approach used in other data warehouse tools: table or partition data is stored in a set of base files, and new, updated, as well as deleted records are stored in incremental files. Each transaction creates a new set of incremental files to change the table or partition. When read, the base files and the incremental files are merged and the changes of the update or deletion are applied.

Writing a transaction table generates some small files in HDFS. Hive provides major and minor compression policies for combining these small files.

Procedure of Automatic Compression

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations** > **MetaStore(Role)** > **Transaction**.

Step 2 Set the following parameters as required:

Table 12-237 Parameter description

Parameter	Description
hive.compactor.check.interval	Interval of executing compression threads. Unit: second. Default value: **300**
hive.compactor.cleaner.run.interval	Interval of executing cleaning threads. Unit: millisecond. Default value: **5,000**.
hive.compactor.delta.num.threshold	Threshold of the number of incremental files that trigger minor compression. Default value: **10**

Parameter	Description
hive.compactor.delta.pct.threshold	Ratio threshold of the total size of incremental files (delta) that trigger Major compression to the size of base files. The value **0.1** indicates that Major compression is triggered when the ratio of the total size of delta files to the size of base files is 10%. Default value: **0.1**
hive.compactor.max.num.delta	Maximum number of incremental files that the compressor will attempt to process in a single job. Default value: **500**
metastore.compactor.initiator.on	Indicates whether to run the startup program thread and cleanup program thread on the MetaStore instance. The value must be **true**. Default value: **false**.
metastore.compactor.worker.threads	Number of compression program work threads running on MetaStore. If this parameter is set to **0**, no compression is performed. To use a transaction, you must set this parameter to a positive number on one or more instances of the MetaStore service. Unit: second Default value: **0**

Step 3 Log in to the Hive client and perform compression. For details, see **Using a Hive Client**.

```
CREATE TABLE table_name (
 id int, name string
)
CLUSTERED BY (id) INTO 2 BUCKETS STORED AS ORC
TBLPROPERTIES ("transactional"="true",
  "compactor.mapreduce.map.memory.mb"="2048",                    -- Specify the properties of a compression map job.
  "compactorthreshold.hive.compactor.delta.num.threshold"="4", -- If there are more than four incremental directories, slight compression is triggered.
  "compactorthreshold.hive.compactor.delta.pct.threshold"="0.5" -- If the ratio of the incremental file size to the basic file size is greater than 50%, deep compression is triggered.
);
```

or

```
ALTER TABLE table_name COMPACT 'minor' WITH OVERWRITE TBLPROPERTIES
("compactor.mapreduce.map.memory.mb"="3072"); -- Specify the properties of a compression map job.
ALTER TABLE table_name COMPACT 'major' WITH OVERWRITE TBLPROPERTIES
("tblprops.orc.compress.size"="8192");        -- Modify any other Hive table attributes.
```

NOTE

After compression, small files are not deleted immediately. After the cleaner thread performs cleaning, the files are deleted in batches.

----End

12.10.29 Switching the Hive Execution Engine to Tez

Scenario

Hive can use the Tez engine to process data computing tasks. Before executing a task, you can manually switch the execution engine to Tez.

Prerequisites

The TimelineServer role of the Yarn service has been installed in the cluster and is running properly.

Switching the Execution Engine on the Client to Tez

Step 1 Install and log in to the Hive client. For details, see **Using a Hive Client**.

Step 2 Run the following commands to switch the engine and enable the **yarn.timeline-service.enabled** parameter:

set hive.execution.engine=tez;

set yarn.timeline-service.enabled=true;

NOTE

- After **yarn.timeline-service.enabled** is enabled, you can view the details about the tasks executed by the Tez engine on TezUI. After this function is enabled, task information will be reported to TimelineServer. If the TimelineServer instance is faulty, the task will fail.
- Tez uses the ApplicationMaster buffer pool. Therefore, **yarn.timeline-service.enabled** must be enabled before Tez tasks are submitted. Otherwise, this parameter cannot take effect and you need to log in to the client again to configure it.
- When the execution engine needs to be switched to another engine, you need to run the **set yarn.timeline-service.enabled=false** command on the client to disable the **yarn.timeline-service.enabled** parameter.
- To specify a Yarn running queue, run the **set tez.queue.name=default** command on the client.

Step 3 Submit and execute the Tez tasks.

Step 4 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Tez** > **TezUI***(host name)* to view the task execution status on the TezUI page.

For versions earlier than MRS 3.x, log in to MRS Manager, choose **Services**, and click **Tez**. On the displayed page, click the link next to **Tez WebUI** to view the task execution status on the TezUI page.

----End

Switching the Default Execution Engine of Hive to Tez

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations** > **HiveServer(Role)**, and search for **hive.execution.engine**.

For versions earlier than MRS 3.x, log in to MRS Manager, choose **Services**, and click **Hive**. On the displayed page, click the **Service Configuration** tab, select **All** from the **Type** drop-down list. On the navigation pane on the left, choose **HiveServer** and search for **hive.execution.engine**.

Step 2 Set **hive.execution.engine** to **tez**.

Step 3 Choose **Hive(Service)** > **Customization** and search for **yarn.site.customized.configs**.

Step 4 Add custom parameter **yarn.timeline-service.enabled** to **yarn.site.customized.configs** and set it to **true**.

NOTE

- After **yarn.timeline-service.enabled** is enabled, you can view the details about the tasks executed by the Tez engine on TezUI. After this function is enabled, task information will be reported to TimelineServer. If the TimelineServer instance is faulty, the task will fail.
- Tez uses the ApplicationMaster buffer pool. Therefore, **yarn.timeline-service.enabled** must be enabled before Tez tasks are submitted. Otherwise, this parameter cannot take effect and you need to log in to the client again to configure it.
- When the execution engine needs to be switched to another one, you need to set the value of parameter **yarn.timeline-service.enabled** to **false**.

Step 5 Click **Save**. In the displayed confirmation dialog box, click **OK**.

For versions earlier than MRS 3.x, click **Save Configuration** and click **Yes** in the displayed dialog box.

Step 6 Choose **Dashboard** > **More** > **Restart Service** to restart the Hive service. Enter the password to restart the service.

For versions earlier than MRS 3.x, Click the **Service Status** tab and choose **More** > **Restart Service** to restart the Hive service.

Step 7 Install and log in to the Hive client. For details, see **Using a Hive Client**.

Step 8 Submit and execute the Tez tasks.

Step 9 Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Tez** > **TezUI***(host name)*. On the displayed TezUI page, view the task execution status.

For versions earlier than MRS 3.x, log in to MRS Manager, choose **Services**, and click **Tez**. On the displayed page, click the link next to **Tez WebUI** to view the task execution status on the TezUI page.

----End

12.10.30 Hive Materialized View

Introduction

A Hive materialized view is a special table obtained based on the query results of Hive internal tables. A materialized view can be considered as an intermediate table that stores actual data and occupies physical space. The tables on which a materialized view depends are called the base tables of the materialized view.

Materialized views are used to pre-compute and save the results of time-consuming operations such as table joining or aggregation. When executing a query, you can rewrite the query statement based on the base tables to the query statement based on materialized views. In this way, you do not need to perform time-consuming operations such as join and group by, thereby quickly obtaining the query result.

NOTE

- A materialized view is a special table that stores actual data and occupies physical space.
- Before deleting a base table, you must delete the materialized view created based on the base table.
- The materialized view creation statement is atomic, which means that other users cannot see the materialized view until all query results are populated.
- A materialized view cannot be created based on the query results of another materialized view.
- A materialized view cannot be created based on the results of a tableless query.
- You cannot insert, update, delete, load, or merge materialized views.
- You can perform complex query operations on materialized views, because they are special tables in nature.
- When the data of a base table is updated, you need to manually update the materialized view. Otherwise, the materialized view will retain the old data. That is, the materialized view expires.
- You can use the describe syntax to check whether the materialized view created based on ACID tables has expired.
- The describe statement cannot be used to check whether a materialized view created based on non-ACID tables has expired.
- A materialized view can store only ORC files. You can use TBLPROPERTIES ('transactional'='true') to create a transactional Hive internal table.

Creating a Materialized View

Syntax

```
CREATE MATERIALIZED VIEW [IF NOT EXISTS] [db_name.]materialized_view_name
  [COMMENT materialized_view_comment]
  DISABLE REWRITE
    [ROW FORMAT row_format]
    [STORED AS file_format]
      | STORED BY 'storage.handler.class.name' [WITH SERDEPROPERTIES (...)]
  ]
  [LOCATION hdfs_path]
  [TBLPROPERTIES (property_name=property_value, ...)]
AS
<query>;
```

NOTE

- Currently, the following materialized view file formats are supported: PARQUET, TextFile, SequenceFile, RCfile, and ORC. If **STORED AS** is not specified in the creation statement, the default file format is ORC.
- Names of materialized views must be unique in the same database. Otherwise, you cannot create a new materialized view, and data files of the original materialized view will be overwritten by the data files queried based on the base table in the new one. As a result, data may be tampered with. (After being tampered with, the materialized view can be restored by re-creating the materialized view.).

Cases

Step 1 Log in to the Hive client and run the following command to enable the following parameters. For details, see **Using a Hive Client**.

set hive.support.concurrency=true;

set hive.exec.dynamic.partition.mode=nonstrict;

set hive.txn.manager=org.apache.hadoop.hive.ql.lockmgr.DbTxnManager;

Step 2 Create a base table and insert data.

```
create table tb_emp(
empno int,ename string,job string,mgr int,hiredate TIMESTAMP,sal float,comm float,deptno int
)stored as orc
tblproperties('transactional'='true');

insert into tb_emp values(7369, 'SMITH', 'CLERK',7902, '1980-12-17 08:30:09',800.00,NULL,20),
(7499, 'ALLEN', 'SALESMAN',7698, '1981-02-20 17:12:00',1600.00,300.00,30),
(7521, 'WARD', 'SALESMAN',7698, '1981-02-22 09:05:34',1250.00,500.00,30),
(7566, 'JONES', 'MANAGER', 7839, '1981-04-02 10:14:13',2975.00,NULL,20),
(7654, 'MARTIN', 'SALESMAN',7698, '1981-09-28 08:36:17',1250.00,1400.00,30),
(7698, 'BLAKE', 'MANAGER',7839, '1981-05-01 11:12:55',2850.00,NULL,30),
(7782, 'CLARK', 'MANAGER',7839, '1981-06-09 15:45:28',2450.00,NULL,10),
(7788, 'SCOTT', 'ANALYST',7566, '1987-04-19 14:05:34',3000.00,NULL,20),
(7839, 'KING', 'PRESIDENT',NULL, '1981-11-17 10:18:25',5000.00,NULL,10),
(7844, 'TURNER', 'SALESMAN',7698, '1981-09-08 09:05:34',1500.00,0.00,30),
(7876, 'ADAMS', 'CLERK',7788, '1987-05-23 15:07:44',1100.00,NULL,20),
(7900, 'JAMES', 'CLERK',7698, '1981-12-03 16:23:56',950.00,NULL,30),
(7902, 'FORD', 'ANALYST',7566, '1981-12-03 08:48:17',3000.00,NULL,20),
(7934, 'MILLER', 'CLERK',7782, '1982-01-23 11:45:29',1300.00,NULL,10);
```

Step 3 Create a materialized view based on the results of the **tb_emp** query.

```
create materialized view group_mv disable rewrite
row format serde 'org.apache.hadoop.hive.serde2.JsonSerDe'
stored as textfile
tblproperties('mv_content'='Total compensation of each department')
as select deptno,sum(sal) sum_sal from tb_emp group by deptno;
```

----End

Applying a Materialized View

Rewrite the query statement based on base tables to the query statement based on materialized views to improve the query efficiency.

Cases

Execute the following query statement:

select deptno,sum(sal) from tb_emp group by deptno having sum(sal)>10000;

Based on the created materialized view, rewrite the query statement:

select deptno, sum_sal from group_mv where sum_sal>10000;

Checking a Materialized View

Syntax

SHOW MATERIALIZED VIEWS [IN database_name]
['identifier_with_wildcards'];

DESCRIBE [EXTENDED | FORMATTED] [db_name.]materialized_view_name;

Cases

show materialized views;

describe formatted group_mv;

Deleting a Materialized View

Syntax

DROP MATERIALIZED VIEW [db_name.]materialized_view_name;

Cases

drop materialized view group_mv;

Rebuilding a Materialized View

When a materialized view is created, the base table data is filled in the materialized view. However, the data that is added, deleted, or modified in the base table is not automatically synchronized to the materialized view. Therefore, you need to manually rebuild the view after updating the data.

Syntax

ALTER MATERIALIZED VIEW [db_name.]materialized_view_name REBUILD;

Cases

alter materialized view group_mv rebuild;

NOTE

When the base table data is updated but the materialized view data is not updated, the materialized view is in the expired state by default.

The describe statement can be used to check whether a materialized view created based on transaction tables has expired. If the value of **Outdated for Rewriting** is **Yes**, the license has expired. If the value of **Outdated for Rewriting** is **No**, the license has not expired.

12.10.31 Hive Log Overview

Log Description

Log path: The default save path of Hive logs is **/var/log/Bigdata/hive/***role name*, the default save path of Hive1 logs is **/var/log/Bigdata/hive1/***role name*, and the others follow the same rule.

- HiveServer: **/var/log/Bigdata/hive/hiveserver** (run log) and **var/log/Bigdata/audit/hive/hiveserver** (audit log)
- MetaStore: **/var/log/Bigdata/hive/metastore** (run log) and **/var/log/Bigdata/audit/hive/metastore** (audit log)
- WebHCat: **/var/log/Bigdata/hive/webhcat** (run log) and **/var/log/Bigdata/audit/hive/webhcat** (audit log)

Log archive rule: The automatic compression and archiving function of Hive is enabled. By default, when the size of a log file exceeds 20 MB (which is adjustable), the log file is automatically compressed. The naming rule of a compressed log file is as follows: <*Original log name*>-<*yyyy-mm-dd_hh-mm-ss*>.[*ID*].**log.zip** A maximum of 20 latest compressed files are reserved. The number of compressed files and compression threshold can be configured.

Table 12-238 Hive log list

Log Type	Log File Name	Description
Run log	/hiveserver/hiveserver.out	Log file that records HiveServer running environment information.
	/hiveserver/hive.log	Run log file of the HiveServer process.
	/hiveserver/hive-omm-*<Date>*-*<PID>*-gc.log.*<No.>*	GC log file of the HiveServer process.
	/hiveserver/prestartDetail.log	Work log file before the HiveServer startup.
	/hiveserver/check-serviceDetail.log	Log file that records whether the Hive service starts successfully
	/hiveserver/cleanupDetail.log	Cleanup log file about the HiveServer uninstallation
	/hiveserver/startDetail.log	Startup log file of the HiveServer process.
	/hiveserver/stopDetail.log	Shutdown log file of the HiveServer process.
	/hiveserver/localtasklog/omm_*<Date>*_*<Task ID>*.log	Run log file of the local Hive task.
	/hiveserver/localtasklog/omm_*<Date>*_*<Task ID>*-gc.log.*<No.>*	GC log file of the local Hive task.
	/metastore/metastore.log	Run log file of the MetaStore process.
	/metastore/hive-omm-*<Date>*-*<PID>*-gc.log.*<No.>*	GC log file of the MetaStore process.
	/metastore/postinstallDetail.log	Work log file after the MetaStore installation.
	/metastore/prestartDetail.log	Work log file before the MetaStore startup
	/metastore/cleanupDetail.log	Cleanup log file of the MetaStore uninstallation
	/metastore/startDetail.log	Startup log file of the MetaStore process.

Log Type	Log File Name	Description
	/metastore/stopDetail.log	Shutdown log file of the MetaStore process.
	/metastore/metastore.out	Log file that records MetaStore running environment information.
	/webhcat/webhcat-console.out	Log file that records the normal start and stop of the WebHCat process.
	/webhcat/webhcat-console-error.out	Log file that records the start and stop exceptions of the WebHCat process.
	/webhcat/prestartDetail.log	Work log file before the WebHCat startup.
	/webhcat/cleanupDetail.log	Cleanup logs generated during WebHCat uninstallation or before WebHCat installation
	/webhcat/hive-omm-<*Date*>-<PID>-gc.log.<*No.*>	GC log file of the WebHCat process.
	/webhcat/webhcat.log	Run log file of the WebHCat process
Audit log	hive-audit.log hive-rangeraudit.log	HiveServer audit log file
	metastore-audit.log	MetaStore audit log file.
	webhcat-audit.log	WebHCat audit log file.
	jetty-<Date>.request.log	Request logs of the jetty service.

Log Levels

Table 12-239 describes the log levels supported by Hive.

Levels of run logs are ERROR, WARN, INFO, and DEBUG from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-239 Log levels

Level	Description
ERROR	Logs of this level record error information about system running.
WARN	Logs of this level record exception information about the current event processing.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of the Yarn service by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level and save the configuration.

NOTE

The Hive log level takes effect immediately after being configured. You do not need to restart the service.

----End

Log Formats

The following table lists the Hive log formats:

Table 12-240 Log formats

Log Type	Format	Example
Run log	<yyyy-MM-dd HH:mm:ss,SSS>| <LogLevel>|<Thread that generates the log>| <Message in the log>| <Location of the log event>	2014-11-05 09:45:01,242 | INFO | main | Starting hive metastore on port 21088 | org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5198)

Log Type	Format	Example
Audit log	<yyyy-MM-dd HH:mm:ss,SSS>| <LogLevel>|<Thread that generates the log>|<User Name><User IP><Time><Operation><Resource><Result><Detail >|< Location of the log event >	2018-12-24 12:16:25,319 | INFO | HiveServer2-Handler-Pool: Thread-185 | UserName=hive UserIP=10.153.2.204 Time=2018/12/24 12:16:25 Operation=CloseSession Result=SUCCESS Detail= | org.apache.hive.service.cli.thrift.ThriftCLIService.logAuditEvent(ThriftCLIService.java:434)

12.10.32 Hive Performance Tuning

12.10.32.1 Creating Table Partitions

Scenario

During the Select query, Hive generally scans the entire table, which is time-consuming. To improve query efficiency, create table partitions based on service requirements and query dimensions.

Procedure

Step 1 For versions earlier than MRS 3.x:

Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click a cluster name. Choose **Nodes** > **Node**. The ECS page is displayed. Click **Remote Login** to log in to the Hive node.

For MRS 3.*x* or later:

Log in to the node where the Hive client has been installed as user **root**.

Step 2 Run the following command to go to the client installation directory, for example, **/opt/client**.

cd /opt/client

Step 3 Run the **source bigdata_env** command to configure environment variables for the client.

Step 4 Run the following command on the client for login:

kinit *Username*

Step 5 Run the following command to log in to the client tool:

beeline

Step 6 Select the static or dynamic partition.

- Static partition:

  Manually enter a partition name, and use the keyword **PARTITIONED BY** to specify partition column name and data type when creating a table. During application development, use the **ALTER TABLE ADD PARTITION** statement to add a partition and use the **LOAD DATA INTO PARTITION** statement to load data to the partition, which supports only static partitions.

- Dynamic partition: Use a query command to insert results to a partition of a table. The partition can be a dynamic partition.

  The dynamic partition can be enabled on the client tool by running the following command:

  set hive.exec.dynamic.partition=true;

  The default mode of the dynamic partition is strict. That is, at least a column must be specified as a static partition, under which dynamic sub-partitions can be created. You can run the following command to enable a completely dynamic partition:

  set hive.exec.dynamic.partition.mode=nonstrict;

NOTE

- The dynamic partition may cause a DML statement to create a large number of partitions and new mapping folders, which deteriorates system performance.
- If there are a large number of files, it takes a long time to run a SQL statement. You can run the **set mapreduce.input.fileinputformat.list-status.num-threads = 100;** statement before running a SQL statement to shorten the time. The parameter **mapreduce.input.fileinputformat.list-status.num-threads** can be set only after being added to the Hive whitelist.

----End

12.10.32.2 Optimizing Join

Scenario

When the Join statement is used, the command execution speed and query speed may be slow in case of large data volume. To resolve this problem, you can optimize Join.

Join optimization can be classified into the following modes:

- Map Join
- Sort Merge Bucket Map Join
- Optimizing Join Sequences

Map Join

Hive Map Join applies to small tables (the table size is less than 25 MB) that can be stored in the memory. The table size can be defined using **hive.mapjoin.smalltable.filesize**, and the default table size is 25 MB.

Map Join has two methods:

- Use /*+ MAPJOIN(join_table) */.

- Set the following parameter before running the statement. The default value is true in the current version.

  set hive.auto.convert.join=true;

There is no Reduce task when Map Join is used. Instead, a MapReduce Local Task is created before the Map job. The task uses TableScan to read small table data to the local computer, saves and writes the data in HashTable mode to a hard disk on the local computer, upload the data to DFS, and saves the data in distributed cache. The small table data that the map task reads from the local disk or distributed cache is the output together with the large table join result.

When using Map Join, make sure that the size of small tables cannot be too large. If small tables use up memory, the system performance will deteriorate and even memory leakage occurs.

Sort Merge Bucket Map Join

The following conditions must be met before using Sort Merge Bucket Map Join:

- The two Join tables are large and cannot be stored in the memory.
- The two tables are bucketed (clustered by (column)) and sorted (sorted by(column)) according to the join key, and the buckets counts of the two tables are in integral multiple relationship.

Set the following parameters to enable Sort Merge Bucket Map Join:

set hive.optimize.bucketmapjoin=true;

set hive.optimize.bucketmapjoin.sortedmerge=true;

This type of Map Join does not have Reduce tasks too. A MapReduce Local Task is started before the Map job to read small table data by bucket to the local computer. The local computer saves the HashTable backup of multiple buckets and writes the backup into HDFS. The backup is also saved in the distributed cache. The small table data that the map task reads from the local disk or distributed cache by bucket is the output after mapping with the large table.

Optimizing Join Sequences

If the Join operation is to be performed on three or more tables and different Join sequences are used, the execution time will be greatly different. Using an appropriate Join sequence can shorten the time for task execution.

Rules of a Join sequence:

- A table with small data volume or a combination with fewer results generated after a Join operation is executed first.
- A table with large data volume or a combination with more results generated after a Join operation is executed later.

For example, the **customer** table has the largest data volume, and fewer results will be generated if a Join operation is performed on the **orders** and **lineitem** tables first.

The original Join statement is as follows.

```
select
  l_orderkey,
```

```
  sum(l_extendedprice * (1 - l_discount)) as revenue,
  o_orderdate,
  o_shippriority
from
  customer,
  orders,
  lineitem
where
  c_mktsegment = 'BUILDING'
  and c_custkey = o_custkey
  and l_orderkey = o_orderkey
  and o_orderdate < '1995-03-22'
  and l_shipdate > '1995-03-22'
limit 10;
```

After the sequence is optimized, the Join statements are as follows:

```
select
  l_orderkey,
  sum(l_extendedprice * (1 - l_discount)) as revenue,
  o_orderdate,
  o_shippriority
from
  orders,
  lineitem,
  customer
where
  c_mktsegment = 'BUILDING'
  and c_custkey = o_custkey
  and l_orderkey = o_orderkey
  and o_orderdate < '1995-03-22'
  and l_shipdate > '1995-03-22'
limit 10;
```

Precautions

Join Data Skew Problem

Data skew refers to the symptom that the task progress is 99% for a long time.

Data skew often exists because the data volume of a few Reduce tasks is much larger than that of others. Most Reduce tasks are complete while a few Reduce tasks are not complete.

To resolve the data skew problem, set **hive.optimize.skewjoin=true** and adjust the value of **hive.skewjoin.key**. **hive.skewjoin.key** specifies the maximum number of keys received by a Reduce task. If the number reaches the maximum, the keys are atomically distributed to other Reduce tasks.

12.10.32.3 Optimizing Group By

Scenario

Optimize the Group by statement to accelerate the command execution and query speed.

During the Group by operation, Map performs grouping and distributes the groups to Reduce; Reduce then performs grouping again. Group by optimization can be performed by enabling Map aggregation to reduce Map output data volume.

Procedure

On a Hive client, set the following parameter:

```
set hive.map.aggr=true
```

Precautions

Group By Data Skew

Group by have data skew problems. When hive.groupby.skewindata is set to true, the created query plan has two MapReduce jobs. The Map output result of the first job is randomly distributed to Reduce tasks, and each Reduce task performs aggregation operations and generates output result. Such processing may distribute the same Group By Key to different Reduce tasks for load balancing purpose. According to the preprocessing result, the second Job distributes Group By Key to Reduce to complete the final aggregation operation.

Count Distinct Aggregation Problem

When the aggregation function count distinct is used in deduplication counting, serious Reduce data skew occurs if the processed value is empty. The empty value can be processed independently. If count distinct is used, exclude the empty value using the where statement and increase the last count distinct result by 1. If there are other computing operations, process the empty value independently and then combine the value with other computing results.

12.10.32.4 Optimizing Data Storage

Scenario

ORC is an efficient column storage format and has higher compression ratio and reading efficiency than other file formats.

You are advised to use **ORC** as the default Hive table storage format.

Prerequisites

You have logged in to the Hive client. For details, see **Using a Hive Client**.

Procedure

- Recommended: **SNAPPY** compression, which applies to scenarios with even compression ratio and reading efficiency requirements.

  Create table *xx (col_name data_type)* stored as orc tblproperties ("orc.compress"="SNAPPY");

- Available: **ZLIB** compression, which applies to scenarios with high compression ratio requirements.

  Create table *xx (col_name data_type)* stored as orc tblproperties ("orc.compress"="ZLIB");

NOTE

xx indicates the specific Hive table name.

12.10.32.5 Optimizing SQL Statements

Scenario

When SQL statements are executed on Hive, if the **(a&b) or (a&c)** logic exists in the statements, you are advised to change the logic to **a & (b or c)**.

Example

If condition a is **p_partkey = l_partkey**, the statements before optimization are as follows:

```
select
	sum(l_extendedprice* (1 - l_discount)) as revenue
from
	lineitem,
	part
where
	(
		p_partkey = l_partkey
		and p_brand = 'Brand#32'
		and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
		and l_quantity >= 7 and l_quantity <= 7 + 10
		and p_size between 1 and 5
		and l_shipmode in ('AIR', 'AIR REG')
		and l_shipinstruct = 'DELIVER IN PERSON'
	)
	or
	(	p_partkey = l_partkey
		and p_brand = 'Brand#35'
		and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
		and l_quantity >= 15 and l_quantity <= 15 + 10
		and p_size between 1 and 10
		and l_shipmode in ('AIR', 'AIR REG')
		and l_shipinstruct = 'DELIVER IN PERSON'
	)
	or
	(	p_partkey = l_partkey
		and p_brand = 'Brand#24'
		and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
		and l_quantity >= 26 and l_quantity <= 26 + 10
		and p_size between 1 and 15
		and l_shipmode in ('AIR', 'AIR REG')
		and l_shipinstruct = 'DELIVER IN PERSON'
	)
```

The statements after optimization are as follows:

```
select
	sum(l_extendedprice* (1 - l_discount)) as revenue
from
	lineitem,
	part
where	p_partkey = l_partkey and
	((
		p_brand = 'Brand#32'
		and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
		and l_quantity >= 7 and l_quantity <= 7 + 10
		and p_size between 1 and 5
		and l_shipmode in ('AIR', 'AIR REG')
		and l_shipinstruct = 'DELIVER IN PERSON'
	)
	or
	(
		p_brand = 'Brand#35'
		and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
```

```
        and l_quantity >= 15 and l_quantity <= 15 + 10
        and p_size between 1 and 10
        and l_shipmode in ('AIR', 'AIR REG')
        and l_shipinstruct = 'DELIVER IN PERSON'
)
or
(
        p_brand = 'Brand#24'
        and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
        and l_quantity >= 26 and l_quantity <= 26 + 10
        and p_size between 1 and 15
        and l_shipmode in ('AIR', 'AIR REG')
        and l_shipinstruct = 'DELIVER IN PERSON'
))
```

12.10.32.6 Optimizing the Query Function Using Hive CBO

Scenario

When joining multiple tables in Hive, Hive supports Cost-Based Optimization (CBO). The system automatically selects the optimal plan based on the table statistics, such as the data volume and number of files, to improve the efficiency of joining multiple tables. Hive needs to collect table statistics before CBO optimization.

NOTE

- The CBO optimizes the joining sequence based on statistics and search criteria. However, the joining sequence may fail to be optimized in some special scenarios, such as data skew occurs and query condition values are not in the table.
- When column statistics collection is enabled, Reduce operations must be performed for aggregation. For insert tasks without the Reduce phase, Reduce operations will be performed to collect statistics.
- This section applies to MRS 3.*x* or later.

Prerequisites

You have logged in to the Hive client. For details, see **Using a Hive Client**.

Procedure

Step 1 On the Manager UI, search for the **hive.cbo.enable** parameter in the service configuration of the Hive component, and select **true** to enable the function permanently.

Step 2 Collect statistics about the existing data in Hive tables manually.

Run the following command to manually collect statistics: Statistics about only one table can be collected. If statistics about multiple tables need to be collected, the command needs to be executed repeatedly.

ANALYZE TABLE [db_name.]tablename [PARTITION(partcol1[=val1], partcol2[=val2], ...)]

COMPUTE STATISTICS

[FOR COLUMNS]

[NOSCAN];

NOTE

- When **FOR COLUMNS** is specified, column-level statistics are collected.
- When NOSCAN is specified, statistics about the file size and number of files will be collected, but specific files will not be scanned.

For example:

analyze table table_name compute statistics;

analyze table table_name compute statistics for columns;

Step 3 Configure the automatic statistics collection function of Hive. After the function is enabled, new statistics will be collected only when you insert data by running the **insert overwrite/into** command.

- Run the following commands on the Hive client to enable the statistics collection function temporarily:

  set hive.stats.autogather = true; enables the automatic collection of table/partition-level statistics.

  set hive.stats.column.autogather = true; enables the automatic collection of column-level statistics.

  NOTE

  - The column-level statistics collection does not support complex data types, such as Map and Struct.
  - The automatic table-level statistics collection does not support Hive on HBase tables.

- On the Manager UI, search for the **hive.stats.autogather** and **hive.stats.column.autogather** parameters in the service configuration of Hive, and select **true** to enable the collection function permanently.

Step 4 Run the following command to view statistics:

DESCRIBE FORMATTED table_name[.column_name] PARTITION partition_spec;

For example:

desc formatted table_name;

desc formatted table_name id;

desc formatted table_name partition(time='2016-05-27');

NOTE

Partition tables only support partition-level statistics collection, so you must specify partitions to query statistics for partition tables.

----End

12.10.33 Common Issues About Hive

12.10.33.1 How Do I Delete UDFs on Multiple HiveServers at the Same Time?

Question

How can I delete permanent user-defined functions (UDFs) on multiple HiveServers at the same time?

Answer

Multiple HiveServers share one MetaStore database. Therefore, there is a delay in the data synchronization between the MetaStore database and the HiveServer memory. If a permanent UDF is deleted from one HiveServer, the operation result cannot be synchronized to the other HiveServers promptly.

In this case, you need to log in to the Hive client to connect to each HiveServer and delete permanent UDFs on the HiveServers one by one. The operations are as follows:

Step 1 Log in to the node where the Hive client is installed as the Hive client installation user.

Step 2 Run the following command to go to the client installation directory:

cd *Client installation directory*

For example, if the client installation directory is **/opt/client**, run the following command:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to authenticate the user:

kinit *Hive service user*

NOTE

The login user must have the Hive admin rights.

Step 5 Run the following command to connect to the specified HiveServer:

beeline -u "jdbc:hive2:// *10.39.151.74*:*21066*/**default;sasl.qop=auth-conf;auth=KERBEROS;principal=***hive/hadoop.<system domain name>@<system domain name>*"

NOTE

- *10.39.151.74* is the IP address of the node where the HiveServer is located.
- *21066* is the port number of the HiveServer. The HiveServer port number ranges from 21066 to 21070 by default. Use the actual port number.
- *hive* is the username. For example, if the Hive1 instance is used, the username is **hive1**.
- You can log in to FusionInsight Manager, choose **System** > **Permission** > **Domain and Mutual Trust**, and view the value of **Local Domain**, which is the current system domain name.
- **hive/hadoop.*<system domain name>*** is the username. All letters in the system domain name contained in the username are lowercase letters.

Step 6 Run the following command to enable the Hive admin rights:

set role admin;

Step 7 Run the following command to delete the permanent UDF:

drop function *function_name*;

NOTE

- *function_name* indicates the name of the permanent function.
- If the permanent UDF is created in Spark, the permanent UDF needs to be deleted from Spark and then from HiveServer by running the preceding command.

Step 8 Check whether the permanent UDFs are deleted from all HiveServers.

- If yes, no further action is required.
- If no, go to **Step 5**.

----End

12.10.33.2 Why Cannot the DROP operation Be Performed on a Backed-up Hive Table?

Question

Why cannot the **DROP** operation be performed for a backed up Hive table?

Answer

Snapshots have been created for an HDFS directory mapping to the backed up Hive table, so the HDFS directory cannot be deleted. As a result, the Hive table cannot be deleted.

When a Hive table is being backed up, snapshots are created for the HDFS directory mapping to the table. The snapshot mechanism of HDFS has the following limitation: If snapshots have been created for an HDFS directory, the directory cannot be deleted or renamed unless the snapshots are deleted. When the **DROP** operation is performed for a Hive table (except the EXTERNAL table), the system attempts to delete the HDFS directory mapping to the table. If the directory fails to be deleted, the system displays a message indicating that the table fails to be deleted.

If you need to delete this table, manually delete all backup tasks related to this table.

12.10.33.3 How to Perform Operations on Local Files with Hive User-Defined Functions

Question

How to perform operations on local files (such as reading the content of a file) with Hive user-defined functions?

Answer

By default, you can perform operations on local files with their relative paths in UDF. The following are sample codes:

```
public String evaluate(String text) {
  // some logic
  File file = new File("foo.txt");
  // some logic
  // do return here
}
```

In Hive, upload the file **foo.txt** used in UDF to HDFS, such as **hdfs://hacluster/tmp/foo.txt**. You can perform operations on the **foo.txt** file by creating UDF with the following sentences:

create function testFunc as 'some.class' using jar 'hdfs://hacluster/somejar.jar', file 'hdfs://hacluster/tmp/foo.txt';

In abnormal cases, if the value of **hive.fetch.task.conversion** is **more**, you can perform operations on local files in UDF by using absolute path instead of relative path. In addition, you must ensure that the file exists on all HiveServer nodes and NodeManager nodes and **omm** user have corresponding operation rights.

12.10.33.4 How Do I Forcibly Stop MapReduce Jobs Executed by Hive?

Question

How do I stop a MapReduce task manually if the task is suspended for a long time?

Answer

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn**.

Step 3 On the left pane, click **ResourceManager(Host name, Active)**, and log in to Yarn.

Step 4 Click the button corresponding to the task ID. On the task page that is displayed, click **Kill Application** in the upper left corner and click **OK** in the displayed dialog box to stop the task.

----End

12.10.33.5 How Do I Monitor the Hive Table Size?

Question

How do I monitor the Hive table size?

Answer

The HDFS refined monitoring function allows you to monitor the size of a specified table directory.

Prerequisites

- The Hive and HDFS components are running properly.
- The HDFS refined monitoring function is normal.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **HDFS** > **Resource**.

Step 3 Click the first icon in the upper left corner of **Resource Usage (by Directory)**, as shown in the following figure.

Step 4 In the displayed sub page for configuring space monitoring, click **Add**.

Step 5 In the displayed **Add a Monitoring Directory** dialog box, set **Name** to the name or the user-defined alias of the table to be monitored and **Path** to the path of the monitored table. Click **OK**. In the monitoring result, the horizontal coordinate indicates the time, and the vertical coordinate indicates the size of the monitored directory.

----End

12.10.33.6 How Do I Prevent Key Directories from Data Loss Caused by Misoperations of the insert overwrite Statement?

Question

How do I prevent key directories from data loss caused by misoperations of the **insert overwrite** statement?

Answer

During monitoring of key Hive databases, tables, or directories, to prevent data loss caused by misoperations of the **insert overwrite** statement, configure **hive.local.dir.confblacklist** in Hive to protect directories.

This configuration item has been configured for directories such as **/opt/** and **/user/hive/warehouse** by default.

Prerequisites

The Hive and HDFS components are running properly.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**, and search for the **hive.local.dir.confblacklist** configuration item.

Step 3 Add paths of databases, tables, or directories to be protected in the parameter value.

Step 4 Click **Save** to save the settings.

----End

12.10.33.7 Why Is Hive on Spark Task Freezing When HBase Is Not Installed?

Scenario

This function applies to Hive.

Perform the following operations to configure parameters. When Hive on Spark tasks are executed in the environment where the HBase is not installed, freezing of tasks can be prevented.

NOTE

The Spark kernel version of Hive on Spark tasks has been upgraded to Spark2x. Hive on Spark tasks can be executed is Spark2x is not installed. If HBase is not installed, when Spark tasks are executed, the system attempts to connect to the ZooKeeper to access HBase until timeout occurs by default. As a result, task freezing occurs.

If HBase is not installed, perform the following operations to execute Hive on Spark tasks. If HBase is upgraded from an earlier version, you do not need to configure parameters after the upgrade.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Configurations** > **All Configurations**.

Step 3 Choose **HiveServer(Role)** > **Customization**. Add a customized parameter to the **spark-defaults.conf** parameter file. Set **Name** to **spark.security.credentials.hbase.enabled**, and set **Value** to **false**.

Step 4 Click **Save**. In the dialog box that is displayed, click **OK**.

Step 5 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Hive** > **Instance**, select all Hive instances, choose **More** > **Restart Instance**, enter the password, and click **OK**.

----End

12.10.33.8 Error Reported When the WHERE Condition Is Used to Query Tables with Excessive Partitions in FusionInsight Hive

Question

When a table with more than 32,000 partitions is created in Hive, an exception occurs during the query with the WHERE partition. In addition, the exception information printed in **metastore.log** contains the following information:

```
Caused by: java.io.IOException: Tried to send an out-of-range integer as a 2-byte value: 32970
    at org.postgresql.core.PGStream.SendInteger2(PGStream.java:199)
    at org.postgresql.core.v3.QueryExecutorImpl.sendParse(QueryExecutorImpl.java:1330)
    at org.postgresql.core.v3.QueryExecutorImpl.sendOneQuery(QueryExecutorImpl.java:1601)
    at org.postgresql.core.v3.QueryExecutorImpl.sendParse(QueryExecutorImpl.java:1191)
    at org.postgresql.core.v3.QueryExecutorImpl.execute(QueryExecutorImpl.java:346)
```

Answer

During a query with partition conditions, HiveServer optimizes the partitions to avoid full table scanning. All partitions whose metadata meets the conditions need to be queried. However, the **sendOneQuery** interface provided by GaussDB limits the parameter value to **32767** in the **sendParse** method. If the number of partition conditions exceeds **32767**, an exception occurs.

12.10.33.9 Why Cannot I Connect to HiveServer When I Use IBM JDK to Access the Beeline Client?

Scenario

When users check the JDK version used by the client, if the JDK version is IBM JDK, the Beeline client needs to be reconstructed. Otherwise, the client will fail to connect to HiveServer.

Procedure

Step 1 Log in to FusionInsight Manager and choose **System** > **Permission** > **User**. In the **Operation** column of the target user, choose **More** > **Download Authentication Credential**, select the cluster information, and click **OK** to download the keytab file.

Step 2 Decompress the keytab file and use WinSCP to upload the decompressed **user.keytab** file to the Hive client installation directory on the node to be operated, for example, **/opt/client**.

Step 3 Run the following command to open the **Hive/component_env** configuration file in the Hive client directory:

vi *Hive client installation directory*/**Hive/component_env**

Add the following content to the end of the line where **export CLIENT_HIVE_URI** is located:

```
\; user.principal=Username @HADOOP.COM\;user.keytab=user.keytab file path/user.keytab
```

----End

12.10.33.10 Description of Hive Table Location (Either Be an OBS or HDFS Path)

Question

Can Hive tables be stored in OBS or HDFS?

Answer

1. The location of a common Hive table stored on OBS can be set to an HDFS path.
2. In the same Hive service, you can create tables stored in OBS and HDFS, respectively.
3. For a Hive partitioned table stored on OBS, the location of the partition cannot be set to an HDFS path. (For a partitioned table stored on HDFS, the location of the partition cannot be changed to OBS.)

12.10.33.11 Why Cannot Data Be Queried After the MapReduce Engine Is Switched After the Tez Engine Is Used to Execute Union-related Statements?

Question

Hive uses the Tez engine to execute union-related statements to write data. After Hive is switched to the MapReduce engine for query, no data is found.

Answer

When Hive uses the Tez engine to execute the union-related statement, the generated output file is stored in the **HIVE_UNION_SUBDIR** directory. After Hive is switched back to the MapReduce engine, files in the directory are not read by default. Therefore, data in the **HIVE_UNION_SUBDIR** directory is not read.

In this case, you can set **mapreduce.input.fileinputformat.input.dir.recursive** to **true** to enable union optimization and determine whether to read data in the directory.

12.10.33.12 Why Does Hive Not Support Concurrent Data Writing to the Same Table or Partition?

Question

Why Does Data Inconsistency Occur When Data Is Concurrently Written to a Hive Table Through an API?

Answer

Hive does not support concurrent data insertion for the same table or partition. As a result, multiple tasks perform operations on the same temporary data directory, and one task moves the data of another task, causing task data exception. The service logic is modified so that data is inserted to the same table or partition in single thread mode.

12.10.33.13 Why Does Hive Not Support Vectorized Query?

Question

When the vectorized parameter **hive.vectorized.execution.enabled** is set to **true**, why do some null pointers or type conversion exceptions occur occasionally when Hive on Tez/MapReduce/Spark is executed?

Answer

Currently, Hive does not support vectorized execution. Many community issues are introduced during vectorized execution and are not resolved stably. The default value of **hive.vectorized.execution.enabled** is **false**. You are advised not to set this parameter to **true**.

12.10.33.14 Why Does Metadata Still Exist When the HDFS Data Directory of the Hive Table Is Deleted by Mistake?

Question

The HDFS data directory of the Hive table is deleted by mistake, but the metadata still exists. As a result, an error is reported during task execution.

Answer

This is a exception caused by misoperation. You need to manually delete the metadata of the corresponding table and try again.

Example:

Run the following command to go to the console:

source ${BIGDATA_HOME}/FusionInsight_BASE_8.1.0.1/install/FusionInsight-dbservice-2.7.0/.dbservice_profile

gsql -p 20051 -U hive -d hivemeta -W HiveUser@

Run the **delete from tbls where tbl_id='xxx';** command.

12.10.33.15 How Do I Disable the Logging Function of Hive?

Question

How do I disable the logging function of Hive?

Answer

Step 1 Log in to the node where the client is installed as user **root**.

Step 2 Run the following command to switch to the client installation directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Log in to the Hive client based on the cluster authentication mode.

- In security mode, run the following command to complete user authentication and log in to the Hive client:

  kinit *Component service user*

  beeline

- In normal mode, run the following command to log in to the Hive client:
  - Run the following command to log in to the Hive client as the component service user:

    beeline -n *component service user*

  - If no component service user is specified, the current OS user is used to log in to the Hive client.

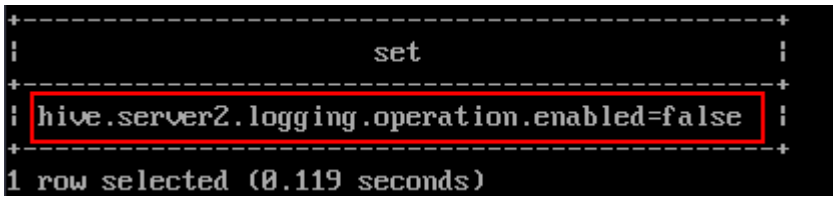
    beeline

Step 5 Run the following command to disable the logging function:

set hive.server2.logging.operation.enabled=false;

Step 6 Run the following command to check whether the logging function is disabled. If the following information is displayed, the logging function is disabled successfully.

set hive.server2.logging.operation.enabled;



----End

12.10.33.16 Why Hive Tables in the OBS Directory Fail to Be Deleted?

Question

In the scenario where the fine-grained permission is configured for multiple MRS users to access OBS, after the permission for deleting Hive tables in the OBS directory is added to the custom configuration of Hive, tables are deleted on the Hive client but still exist in the OBS directory.

Answer

You do not have the permission to delete directories on OBS. As a result, Hive tables cannot be deleted. In this case, modify the custom IAM policy of the agency and configure Hive with the permission for deleting tables in the OBS directory.

12.10.33.17 Hive Configuration Problems

- The error message "java.lang.OutOfMemoryError: Java heap space." is displayed during Hive SQL execution.

  Solution:

  - For MapReduce tasks, increase the values of the following parameters:

    set mapreduce.map.memory.mb=8192;

    set mapreduce.map.java.opts=-Xmx6554M;

    set mapreduce.reduce.memory.mb=8192;

    set mapreduce.reduce.java.opts=-Xmx6554M;
  - For Tez tasks, increase the value of the following parameter:

    set hive.tez.container.size=8192;
- After a column name is changed to a new one using the Hive SQL **as** statement, the error message "Invalid table alias or column reference 'xxx'." is displayed when the original column name is used for compilation.

  Solution: Run the **set hive.cbo.enable=true;** statement.
- The error message "Unsupported SubQuery Expression 'xxx': Only SubQuery expressions that are top level conjuncts are allowed." is displayed during Hive SQL subquery compilation.

  Solution: Run the **set hive.cbo.enable=true;** statement.
- The error message "CalciteSubquerySemanticException [Error 10249]: Unsupported SubQuery Expression Currently SubQuery expressions are only allowed as Where and Having Clause predicates." is displayed during Hive SQL subquery compilation.

  Solution: Run the **set hive.cbo.enable=true;** statement.
- The error message "Error running query: java.lang.AssertionError: Cannot add expression of different type to set." is displayed during Hive SQL compilation.

  Solution: Run the **set hive.cbo.enable=false;** statement.
- The error message "java.lang.NullPointerException at org.apache.hadoop.hive.ql.udf.generic.GenericUDAFComputeStats $GenericUDAFNumericStatsEvaluator.init." is displayed during Hive SQL execution.

  Solution: Run the **set hive.map.aggr=false;** statement.
- When **hive.auto.convert.join** is set to **true** (enabled by default) and **hive.optimize.skewjoin** is set to **true**, the error message "ClassCastException org.apache.hadoop.hive.ql.plan.ConditionalWork cannot be cast to org.apache.hadoop.hive.ql.plan.MapredWork" is displayed.

  Solution: Run the **set hive.optimize.skewjoin=false;** statement.
- When **hive.auto.convert.join** is set to **true** (enabled by default), **hive.optimize.skewjoin** is set to **true**, and **hive.exec.parallel** is set to **true**, the error message "java.io.FileNotFoundException: File does not exist:xxx/reduce.xml" is displayed.

  Solution:

  - Method 1: Switch the execution engine to Tez. For details, see **Switching the Hive Execution Engine to Tez**.
  - Method 2: Run the **set hive.exec.parallel=false;** statement.

  - Method 3: Run the **set hive.auto.convert.join=false;** statement.
- Eerror message "NullPointerException at org.apache.hadoop.hive.ql.exec.CommonMergeJoinOperator.mergeJoinComputeKeys" is displayed when Hive on Tez executes bucket map join.

  Solution: Run the **set tez.am.container.reuse.enabled=false;** statement.

12.11 Using Hue (Versions Earlier Than MRS 3.x)

12.11.1 Using Hue from Scratch

Hue provides the file browser function using a graphical user interface (GUI) so that you can view files and directories on Hive.

Prerequisites

You have installed Hive and Hue, and the Kerberos authentication cluster in the running state.

Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Open the Hue web UI and choose **Query Editors** > **Hive**.

Step 3 In **Databases**, select a Hive database, the default database is **default**.

The system displays all available tables. You can enter a keyword of the table name to search for the desired table.

Step 4 Click the desired table name. All columns in the table are displayed.

Step 5 Enter the HiveQL statements in the area for editing.

create table hue_table(id int,name string,company string) row format delimited fields terminated by ',' stored as textfile;

Click  and select **Explain**. The editor checks the syntax and execution plan of the entered HiveQL statements. If the statements have syntax errors, the editor reports **Error while compiling statement**.

Step 6 Click , and select the engine for executing the HiveQL statements.

Step 7 Click  to execute the HiveQL statements.

Step 8 In the command text box, enter **show tables;** and click  . Check whether the **hue-table** table created in **Step 5** exists in the result.

----End

12.11.2 Accessing the Hue Web UI

Scenario

After Hue is installed in an MRS cluster, users can use Hadoop and Hive on the Hue web UI.

This section describes how to open the Hue web UI on the MRS cluster.

NOTE

To access the Hue web UI, you are advised to use a browser that is compatible with the Hue WebUI, for example, Google Chrome 50. The Internet Explorer may be incompatible with the Hue web UI.

Impact on the System

Site trust must be added to the browser when you access Manager and Hue web UI for the first time. Otherwise, the Hue web UI cannot be accessed.

Prerequisites

When Kerberos authentication is enabled, the MRS cluster administrator has assigned the permission for using Hive to the user. For details, see "MRS Manager Operation Guide" > "Permission Management" > "Creating a User" in *MapReduce Service User Guide*. For example, create a human-machine user named **hueuser**, add the user to user groups **hive** (the primary group), **hadoop**, and **supergroup**, and role **System_administrator**.

This user is used to log in to the Hue WebUI.

Procedure

Step 1 Log in to the service page, click the cluster name to go to the cluster details page, and choose **Components**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 2 Select **Hue**. On the right side of **Hue WebUI**, click the link to log in to the Hue web UI as user **hueuser**.

Hue WebUI provides the following functions:

- If Hive is installed in the MRS cluster, you can use **Query Editors** to execute query statements of Hive. Hive has been installed in the MRS cluster.
- If Hive is installed in the MRS cluster, you can use **Data Browsers** to manage Hive tables.
- If HDFS is installed in the MRS cluster, you can use  to view directories and files in HDFS.
- If Yarn is installed in the MRS cluster, you can use  to view all jobs in the MRS cluster.

NOTE

- When you log in to the Hue web UI as user **hueuser** for the first time, you need to change the password.
- After obtaining the URL for accessing the Hue web UI, you can give the URL to other users who cannot access MRS Manager for accessing the Hue web UI.
- If you perform operations on the Hue WebUI only but not on Manager, you must enter the password of the current login user when accessing Manager again.

----End

12.11.3 Hue Common Parameters

Navigation Path

For details about how to set parameters, see **Modifying Cluster Service Configuration Parameters**.

Parameters

Table 12-241 Hue common parameters

Parameter	Description	Default Value	Value Range
HANDLER_ACCESSLOG_LEVEL	Hue access log level	DEBUG	• ERROR • WARN • INFO • DEBUG
HANDLER_AUDITSLOG_LEVEL	Hue audit log level	DEBUG	• ERROR • WARN • INFO • DEBUG
HANDLER_ERRORLOG_LEVEL	Hue error log level	ERROR	• ERROR • WARN • INFO • DEBUG
HANDLER_LOGFILE_LEVEL	Hue run log level	INFO	• ERROR • WARN • INFO • DEBUG
HANDLER_LOGFILE_MAXBACKUPINDEX	Maximum number of Hue log files.	20	1 to 999
HANDLER_LOGFILE_SIZE	Maximum size of a Hue log file.	5 MB	-

12.11.4 Using HiveQL Editor on the Hue Web UI

Scenario

Users can use the Hue web UI to execute HiveQL statements in a cluster.

Accessing Query Editors

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Choose **Query Editors** > **Hive**. The **Hive** page is displayed.

Hive supports the following functions:

- Executes and manages HiveQL statements.
- View the HiveQL statements saved by the current user in **Saved Queries**.
- Query HiveQL statements executed by the current user in **Query History**.
- Click  to display all databases included in **Databases** of Hive.

----End

Executing HiveQL Statements

Step 1 Choose **Query Editors > Hive**. The **Hive** page is displayed.

Step 2 Click  and select a database from **Databases**. The default database is **default**.

The system displays all available tables in the database. You can enter a keyword of the table name to search for the desired table.

Step 3 Click the desired table name. All columns in the table are displayed.

Move the cursor to the row of the table and click  . Column details are displayed.

Step 4 Enter the query statements in the area for editing HiveQL statements.

Click  and select **Explain**. The editor checks the syntax and execution plan of the entered statements. If the statements have syntax errors, the editor reports **Error while compiling statement**.

Step 5 Click and select the engine for executing the HiveQL statements.

- **mr**: MapReduce computing framework
- **spark**: Spark computing framework
- **tez**: Tez computing framework

  NOTE

  Tez is applicable to MRS 1.9.*x* and later versions.

Step 6 Click  to execute the HiveQL statements.

NOTE

- If you want to use the entered HiveQL statements again, click to save them.
- To format HiveQL statements, click and select **Format**.
- To delete an entered HiveQL statement, click and select **Clear**.
- Clear the entered statement and execute a new statement. Click and select **New query**.
- Viewing history:

  Click **Query History** to view the HiveQL running status. You can view the history of all the statements or only the saved statements. If many historical records exist, you can enter keywords in the text box to search for desired records.
- Advanced query configuration:

  Click in the upper right corner to configure information such as files, functions, and settings.
- Viewing the information of shortcut keys:

  Click in the upper right corner to view all shortcut keys.

----End

Viewing Execution Results

Step 1 In the **Hive** execution area, **Query History** is displayed by default.

Step 2 Click **Results** to view the execution result of the executed statement.

----End

Managing Query Statements

Step 1 Choose **Query Editors > Hive**. The **Hive** page is displayed.

Step 2 Click **Saved Queries**.

Click a saved statement. The system automatically adds the statement to the editing area.

----End

Modifying Query Editors Settings

Step 1 On the **Hive** tab page, click .

Step 2 Click on the right of **Files** and click to specify the directory for storing the file.

You can click to add a file resource.

Step 3 Click on the right of **Functions** and enter the names of user-defined function and function class.

You can click  to add a customized function.

Step 4 Click  on the right of **Settings**, enter the Hive parameter name in the **Key**, and value in **Value**. The current Hive session connects to Hive based on the customized configuration.

You can click  to add a parameter.

----End

12.11.5 Using the Metadata Browser on the Hue Web UI

Scenario

Users can use the Hue web UI to manage Hive metadata in an MRS cluster.

Using Metastore Manager

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Choose **Data Browsers** > **Metastore Tables**, and access **Metastore Manager**.

- Viewing metadata of Hive tables

  In the left navigation pane, move the cursor to a table and click  on the right. The metadata of the Hive table is displayed.

- Managing metadata of Hive tables

  On the metadata page of a Hive table, you can click  in the upper right corner to import data, click  to browse data, and click  to view the location of the table file.

  ⚠ CAUTION

  The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

- Managing Hive metadata tables

  Click  in the upper right corner to create a table in the database based on the uploaded files. Or click  in the upper right corner to manually create a table.

Accessing Metastore Manager

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Choose **Data Browsers** > **Metastore Tables**, and access **Metastore Manager**.

Metastore Manager supports the following functions:

- Creating a Hive table from a file
- Manually creating a Hive table
- Viewing Hive table metadata

----End

Creating a Hive table from a File

Step 1 Access **Metastore Manager** and select a database in **Databases**.

The default database is **default**.

Step 2 Click  . The **Create a new table from a file** page is displayed.

Step 3 Select a file.

1. In **Table Name**, enter a Hive table name.

   A Hive table name contains no more than 128 characters, including letters, numbers, or underscores (_), and must start with a letter or number.
2. In **Description**, enter description about the Hive table as required.
3. In **Input File or Location**, click and select a Hive table file from HDFS. The file is used to store new data of the Hive table.

   If the file is not stored in HDFS, click **Upload a file** to upload the file from the local directory to HDFS. Multiple files can be simultaneously uploaded. The files cannot be empty.
4. If you need to import the data in the file to the Hive table, select **Import data** as **Load method**. By default, **Import data** is selected.

   If you select **Create External Table**, a Hive external table is created.

   NOTE

   If you select **Create External Table**, set **Input File or Location** to a path.

   If you select **Leave Empty**, an empty Hive table is created.
5. Click **Next**.

Step 4 Set a delimiter.

1. In **Delimiter**, select one.

   If your desired delimiter is not in the list, select **Other..** and enter a delimiter.
2. Click **Preview** to preview data processing.
3. Click **Next**.

Step 5 Define a column.

1. If you click on the right side of **Use first row as column names**, the first row of data in the file is used as a column name. If you do not click it, the first row of data is not used as the column name.
2. In **Column name**, set a name for each column.

   A Hive table name contains no more than 128 characters, including letters, numbers, or underscores (_), and must start with a letter or number.

   NOTE

   You can rename columns in batches by clicking on the right side of **Bulk edit column names**. Enter all column names and separate them by commas (,).
3. In **Column Type**, select a type for each column.

Step 6 Click **Create Table** to create the table. Wait for Hue to display information about the Hive table.

----End

Manually Creating a Hive Table

Step 1 Access **Metastore Manager** and select a database in **Databases**.

The default database is **default**.

Step 2 Click . The **Create a new table manually** page is displayed.

Step 3 Set a table name.

1. In **Table Name**, enter a Hive table name.

   A Hive table name contains no more than 128 characters, including letters, numbers, or underscores (_), and must start with a letter or number.
2. In **Description**, enter description about the Hive table as required.
3. Click **Next**.

Step 4 Select a data storage format.

- If data needs to be separated by delimiters, select **Delimited** and perform **Step 5**.
- If data needs to be stored in serialization format, select **SerDe** and perform **Step 6**.

Step 5 Set a delimiter.

1. In **Field terminator**, set a column delimiter.

   If your desired delimiter is not in the list, select **Other..** and enter a delimiter.
2. In **Collection terminator**, set a delimiter to separate the data set of columns of the **array** type in Hive. For example, the type of a column is array. A value needs to store **employee** and **manager**. The user specifies a colon (**:**) as the delimiter. Therefore, the final value is **employee:manager**.
3. In **Map key terminator**, set a delimiter to separate the data set of columns of the **map** type in Hive. For example, the type of a column is map. A value needs to store **home** of **aaa** and **company** of **bbb**. The user defines **|** as the delimiter. Therefore, the final value is **home|aaa:company|bbb**.

4. Click **Next** and perform **Step 7**.

Step 6 Set serialization properties.

1. In **SerDe Name**, enter the class name of the serialization format: **org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe**

   Users can expand Hive to support more customized serialization classes.
2. In **Serde properties**, enter the value of the serialization format: **"field.delim"="," "colelction.delim"=":" "mapkey.delim"="|"**
3. Click **Next** and perform **Step 7**.

Step 7 Select a data table format and click **Next**.

- **TextFile**: indicates that data is stored in text files.
- **SequenceFile**: indicates that data is stored in binary files.
- **InputFormat**: indicates that data in files is used in the customized input and output formats.

  Users can expand Hive to support more customized formatting classes.

  a. In **InputFormat Class**, enter the class used by input data: **org.apache.hadoop.hive.ql.io.RCFileInputFormat**

  b. In **OutputFormat Class**, enter the class used by output data: **org.apache.hadoop.hive.ql.io.RCFileOutputFormat**

Step 8 Select a file storage location and click **Next**.

Use default location is selected by default. If you want to customize a storage location, deselect the default value and specify a file storage location in **External location** by clicking .

Step 9 Set columns of the Hive table.

1. In **Column name**, set a column name.

   A Hive table name contains no more than 128 characters, including letters, numbers, or underscores (_), and must start with a letter or number.
2. In **Column type**, select a type for each column.

   Click **Add a column** to add a new column.
3. Click **Add a partition** to add a new partition for the Hive table to improve the query efficiency.

Step 10 Click **Create Table** to create a new table. Wait for Hue to display information about the Hive table.

----End

Managing the Hive Table

Step 1 Access **Metastore Manager** and select a database in **Databases**. All tables in the database are displayed on the page.

The default database is **default**.

Step 2 Click a table name in the database to view table details.

The following operations are supported: importing data, browsing data,, or viewing file storage location. When viewing all tables in the database, you can

select tables and perform the following operations such as viewing tables and browsing data.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

----End

12.11.6 Using File Browser on the Hue Web UI

Scenario

Users can use the Hue web UI to manage files in HDFS in a cluster.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

File Browser (File Browser)

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click  . The **File Browser** page is displayed.

You can view the home directory of the current login user.

On the **File Browser** page, the following information about subdirectories for files in the directory is displayed.

Table 12-242 HDFS file attributes

Attribute	Description
Name	Name of a directory or file
Size	File size
User	Owner of a directory or file
Group	Group of a directory or file

Attribute	Description
Permissions	Permission of a directory or file
Date	Time when a directory or file is created

Step 3 In the search box, enter a keyword. The system automatically searches directories or files in the current directory.

Step 4 Clear the search criteria. The system displays all directories or files.

----End

Performing Actions

Step 1 Click  and select one or more directories or files.

Step 2 Click **Actions**. On the menu that is displayed, select an operation.

- **Rename**: renames a directory or file.
- **Move**: moves a file. In **Move to**, select a new directory and click **Move**.
- **Copy**: copies the selected files or directories.
- **Change permissions**: changes permission to access the selected directory or file.
  - You can grant the owner, the group, or other users with the **Read**, **Write**, and **Execute** permissions.
  - **Sticky**: indicates that only HDFS administrators, directory owners, and file owners can move files in the directory.
  - **Recursive**: indicates that permission is granted to subdirectories recursively.
- **Storage policies**: indicates the policies for storing files or directories in HDFS.
- **Summary**: indicates that you can view HDFS storage information about the selected file or directory.

----End

Accessing Other Directories

Step 1 Click the directory name, type a full path you want to access, for example, **/mr-history/tmp**, and press **Enter**.

The current user must have permission to access other directories.

Step 2 Click **Home** to go to the home directory.

Step 3 Click **History**. The history records of directory access are displayed and the directories can be accessed again.

Step 4 Click **Trash** to access the recycle bin of the current directory.

Click **Empty Trash** to clean up the recycle bin.

----End

Uploading User Files

Step 1 Click  and click **Upload**.

Step 2 Select an operation.

- **Files**: uploads user files to the current user.
- **Zip/Tgz/Bz2 file**: uploads a compressed file. In the dialog box that is displayed, click **Select ZIP, TGZ or BZ2 files** to select the compressed file to be uploaded. The system automatically decompresses the file in HDFS. Compressed files in **ZIP**, **TGZ**, and **BZ2** formats are supported.

----End

Creating a New File or Directory

Step 1 Click  and click **New**.

Step 2 Select an operation.

- **File**: creates a file. Enter a file name and click **Create**.
- **Directory**: creates a directory. Enter a directory name and click **Create**.

----End

Storage Policy Definition and Usage

NOTE

If the value of Hue parameter **fs_defaultFS** is set to **viewfs://ClusterX**, the big data storage policy cannot be enabled.

Step 1 Log in to MRS Manager.

Step 2 On MRS Manager, choose **System** > **Permission > Manage Role** > **Create Role**.

1. Set **Role Name**.
2. Choose **Configure Resource Permission > Hue**, select **Storage Policy Admin**, and click **OK** to grant the storage policy administrator permission to the role.

Step 3 Choose **System** > **Permission** > **Manage User Group** > **Create User Group**, set **Group Name**, and click **Select and Add Role** next to **Role**. On the displayed page, select the created role and click **OK** to add the role to the group.

Step 4 Choose **System** > **Permission** > **Manage User** > **Create User**.

1. Specify the **Username** of a user who can log in to the Hue web UI and has the **Storage Policy Admin** permission.
2. Set **User Type** to **Human-machine**.
3. Set **Password** and **Confirm Password** for logging in to the Hue web UI.
4. Click **Select and Join User Group** next to **User Group**. On the page that is displayed, select the created user group, **supergroup**, **hadoop**, and **hive**, and click **OK**.
5. Set **Primary Group** to **hive**.

6. Click **Select and Add Role** on the right of **Assign Rights by Role**. On the Select snf page that is displayed, select the newly created role and the **System_administrator** role, and click **OK**.
7. Click **OK**. The user is added successfully.

Step 5 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 6 Click  in the upper right corner.

Step 7 Select the check box of the directory and click **Action** on the upper part of the page. Then select **Storage policies**.

Step 8 In the dialog box that is displayed, set a new storage policy and click **OK**.

----End

12.11.7 Using Job Browser on the Hue Web UI

Scenario

You can use the Hue web UI to query all jobs in the cluster.

Accessing Job Browser

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click **Job Browser**.

View the jobs in the cluster.

NOTE

The number on **Job Browser** indicates the total number of jobs in the cluster.

Job Browser displays the following job information.

Table 12-243 MRS job attributes

Attribute	Description
Logs	Log information. If a job has logs, is displayed.
ID	Job ID, which is generated by the system automatically.
Name	Job name
Application Type	Job type
Status	Job status. Possible values are **RUNNING**, **SUCCEEDED**, **FAILED**, and **KILLED**.
User	User who starts the job
Maps	Map progress
Reduces	Reduce progress

Attribute	Description
Queue	Yarn queue used for job running
Priority	Job running priority
Duration	Job running duration
Submitted	Time when the job is submitted to the MRS cluster

NOTE

If the MRS cluster has Spark, the **Spark-JDBCServer** job is started by default to execute tasks.

----End

Searching for Jobs

Step 1 Enter keywords in **Username** or **Text** on the **Job Browser** page to search for the desired jobs.

Step 2 Clear the search criteria. The system displays all jobs.

----End

Querying Job Details

Step 1 In the job list on the **Job Browser** page, click the row that contains the desired job to view details.

Step 2 On the **Metadata** tab page, you can view the metadata of the job.

NOTE

You can click to open job running logs.

----End

12.12 Using Hue (MRS 3.*x* or Later)

12.12.1 Using Hue from Scratch

Hue aggregates interfaces which interact with most Apache Hadoop components and enables you to use Hadoop components with ease on a web UI. You can operate components such as HDFS, Hive, HBase, Yarn, MapReduce, Oozie, and Spark SQL on the Hue web UI.

Prerequisites

You have installed Hue, and the Kerberos authentication cluster is in the running state.

Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the navigation tree on the left, click the editor icon  and choose **Hive**.

Step 3 Select a Hive database from the **Database** drop-down list box. The default database is **default**.

The system displays all available tables. You can enter a keyword of the table name to search for the desired table.

Step 4 Click the desired table name. All columns in the table are displayed.

Step 5 Enter the HiveQL statements in the area for editing.

create table hue_table(id int,name string,company string) row format delimited fields terminated by ',' stored as textfile;

Step 6 Click  to execute the HiveQL statements.

Step 7 In the command text box, enter **show tables;** and click  . Check whether the **hue_table** table created in **Step 5** exists in the **Result**.

----End

12.12.2 Accessing the Hue Web UI

Scenario

After Hue is installed in an MRS cluster, users can use Hadoop-related components on the Hue web UI.

This section describes how to open the Hue web UI on the MRS cluster.

NOTE

To access the Hue web UI, you are advised to use a browser that is compatible with the Hue WebUI, for example, Google Chrome 50. The Internet Explorer may be incompatible with the Hue web UI.

Impact on the System

Site trust must be added to the browser when you access Manager and Hue web UI for the first time. Otherwise, the Hue web UI cannot be accessed.

Prerequisites

When Kerberos authentication is enabled, the MRS cluster administrator has assigned the permission for using Hive to the user. For details, see . For example, create a human-machine user named **hueuser**, add the user to user groups **hive** (the primary group), **hadoop**, **supergroup**, and **System_administrator**, and assign the **System_administrator** role.

This user is used to log in to Manager.

Procedure

Step 1 Log in to the service page.

For versions earlier than MRS 3.*x*, click the cluster name on the MRS console and choose **Components** > **Hue**.

For MRS 3.*x* or later, log in to FusionInsight Manager (for details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**) and choose **Cluster** > **Services** > **Hue**.

Step 2 On the right of **Hue WebUI**, click the link to open the Hue web UI.

Hue WebUI provides the following functions:

- Click  to execute query statements of Hive and SparkSQL as well as Notebook code. Make sure that Hive and Spark2x have been installed in the MRS cluster before this operation.
- Click  to submit workflow tasks, scheduled tasks, and bundle tasks.
- Click  to view, import, and export tasks on the Hue web UI, such as workflow tasks, scheduled tasks, and bundle tasks.
- Click  to manage metadata in Hive and SparkSQL. Make sure that Hive and Spark2x have been installed in the MRS cluster before this operation.
- Click  to view the directories and files in HDFS. Make sure that HDFS has been installed in the MRS cluster before this operation.
- Click  to view all jobs in the MRS cluster. Make sure that Yarn has been installed in the MRS cluster before this operation.
- Use  to create or query HBase tables. Make sure that the HBase component has been installed in the MRS cluster and the Thrift1Server instance has been added before this operation.
- Use  to import data that is in the CSV or TXT format.

NOTE

- When you log in to the Hue web UI as user **hueuser** for the first time, you need to change the password.
- After obtaining the URL for accessing the Hue web UI, you can give the URL to other users who cannot access MRS Manager for accessing the Hue web UI.
- If you perform operations on the Hue WebUI only but not on Manager, you must enter the password of the current login user when accessing Manager again.

----End

12.12.3 Hue Common Parameters

Page Access

Go to the **All Configurations** page of the Hue service by referring to **Modifying Cluster Service Configuration Parameters**.

Parameter Description

For details about Hue common parameters, see **Table 12-244**.

Table 12-244 Hue common parameters

Configuration	Description	Default Value	Value Range
HANDLER_ACCESSLOG_LEVEL	Hue access log level.	DEBUG	• ERROR • WARN • INFO • DEBUG
HANDLER_AUDITSLOG_LEVEL	Hue audit log level.	DEBUG	• ERROR • WARN • INFO • DEBUG
HANDLER_ERRORLOG_LEVEL	Hue error log level.	ERROR	• ERROR • WARN • INFO • DEBUG
HANDLER_LOGFILE_LEVEL	Hue run log level.	INFO	• ERROR • WARN • INFO • DEBUG
HANDLER_LOGFILE_MAXBACKUPINDEX	Maximum number of Hue log files.	20	1 to 999
HANDLER_LOGFILE_SIZE	Maximum size of a Hue log file.	5 MB	-

12.12.4 Using HiveQL Editor on the Hue Web UI

Scenario

Users can use the Hue web UI to execute HiveQL statements in an MRS cluster.

Access Editor

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the navigation tree on the left, click  and choose **Hive**. The **Hive** page is displayed.

Hive supports the following functions:

- Executes and manages HiveQL statements.
- Views the HiveQL statements saved by the current user in **Saved Queries**.
- Queries HiveQL statements executed by the current user in **Query History**.

----End

Executing HiveQL Statements

Step 1 Select a Hive database from the **Database** drop-down list box. The default database is **default**.

The system displays all available tables. You can enter a keyword of the table name to search for the desired table.

Step 2 Click the desired table name. All columns in the table are displayed.

Move the cursor to the row where the table or column is located and click . Column details are displayed.

Step 3 Enter the query statements in the area for editing HiveQL statements.

Step 4 Click to execute the HiveQL statements.

NOTE

- If you want to use the entered HiveQL statements again, click to save them.
- Advanced query configuration:

  Click in the upper right corner to configure information such as files, functions, and settings.
- Viewing the information of shortcut keys:

  Click in the upper right corner to view the syntax and keyboard shortcut information.
- To delete an entered HiveQL statement, click the triangle next to and select **Clear**.
- Viewing history:

  Click **Query History** to view the HiveQL running status. You can view the history of all the statements or only the saved statements. If many historical records exist, you can enter keywords in the text box to search for desired records.

----End

Viewing Execution Results

Step 1 View the execution results below the execution area on **Hive**. The **Query History** tab page is displayed by default.

Step 2 Click a result to view the execution result of the executed statement.

----End

Managing Query Statements

Step 1 Click **Saved Queries**.

Step 2 Click a saved statement. The system automatically adds the statement to the editing area.

----End

Modifying the Session Configuration of the Hue Editor

Step 1 On the editor page, click .

Step 2 Click on the right of **Files**, and then click to select files.

You can click next to **Files** to add a file resource.

Step 3 In the **Functions** area, enter a user-defined name and the class name of the function.

You can click next to **Functions** to add a customized function.

Step 4 In the **Settings** area, enter the Hive parameter name in the **Key**, and value in **Value**. The current Hive session connects to Hive based on the customized configuration.

You can click to add a parameter.

----End

12.12.5 Using the SparkSql Editor on the Hue Web UI

Scenario

You can use Hue to execute SparkSql statements in a cluster on a graphical user interface (GUI).

Configuring Spark2x

Before using the SparkSql editor, you need to modify the Spark2x configuration.

Step 1 Go to the Spark2x configuration page. For details, see **Modifying Cluster Service Configuration Parameters**.

Step 2 Set the Spark2x multi-instance mode. Search for and modify the following parameters of the Spark2x service:

Parameter	Value
spark.thriftserver.proxy.enabled	false
spark.scheduler.allocation.file	#{conf_dir}/fairscheduler.xml

Step 3 Go to the JDBCServer2x customization page and add the following customized items to the **spark.core-site.customized.configs** parameter:

Set **hadoop.proxyuser.hue.groups** to *.

Set **hadoop.proxyuser.hue.hosts** to *.

Step 4 Save the configuration and restart the meta and Spark2x services.

----End

Accessing the Editor

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the navigation tree on the left, click  and choose **SparkSql**. The **SparkSql** page is displayed.

SparkSql supports the following functions:

- Executes and manages SparkSql statements.
- Views the SparkSql statements saved by the current user in **Saved Queries**.
- Queries SparkSql statements executed by the current user in **Query History**.

----End

Executing SparkSql Statements

Step 1 Select a SparkSql database from the **Database** drop-down list box. The default database is **default**.

The system displays all available tables. You can enter a keyword of the table name to search for the desired table.

Step 2 Click the desired table name. All columns in the table are displayed.

Move the cursor to the row of the table and click  . Column details are displayed.

Step 3 In the SparkSql statement editing area, enter the query statement.

Click the triangle next to  and select **Explain**. The editor checks the syntax and execution plan of the entered statements. If the statements have syntax errors, the editor reports **Error while compiling statement**.

Step 4 Click  to execute the SparkSql statement.

NOTE

- If you want to use the entered SparkSql statements again, click to save them.
- Advanced query configuration:

  Click in the upper right corner to configure information such as files, functions, and settings.
- Viewing the information of shortcut keys:

  Click in the upper right corner to view the syntax and keyboard shortcut information.
- To format the SparkSql statement, click the triangle next to and select **Format**.
- To delete an entered SparkSql statement, click the triangle next to and select **Clear**.
- Viewing historical records:

  Click **Query History** to view the SparkSql running status. You can view the history of all the statements or only the saved statements. If many historical records exist, you can enter keywords in the text box to search for desired records.

----End

Viewing Execution Results

Step 1 View the execution results below the execution area on **SparkSql**. The **Query History** tab page is displayed by default.

Step 2 Click a result to view the execution result of the executed statement.

----End

Managing Query Statements

Step 1 Click **Saved Queries**.

Step 2 Click a saved statement. The system automatically adds the statement to the editing area.

----End

12.12.6 Using the Metadata Browser on the Hue Web UI

Scenario

Users can use the Hue web UI to manage Hive metadata in an MRS cluster.

Using Metadata Manager

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

- Viewing metadata of Hive tables

  Click in the navigation tree on the left and click a table name. The metadata of the Hive table is displayed.

- Managing metadata of Hive tables

  On the metadata information page of a Hive table:

  - Click **Import** in the upper right corner to import data.
  - Click **Overview** to view the location of the table file in the **PROPERTIES** field.

    View the field information of each column in a Hive table and manually add description information. Note that the added description information is not the field comments in the Hive table.
  - Click **Sample** to browse data.
- Managing Hive metadata tables

  Click ➕ in the left list to create a table based on the uploaded file in the database. You can also manually create a table.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

12.12.7 Using File Browser on the Hue Web UI

Scenario

Users can use the Hue web UI to manage files in HDFS.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

Accessing File Browser

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the left navigation pane, click  . The **File Browser** page is displayed.

By default, the homepage of **File Browser** is the home directory of the current login user. On the displayed page, the following information about subdirectories for files in the directory is displayed:

Table 12-245 HDFS file attributes

Attribute	Description
Name	Name of a directory or file
Size	File size
User	Owner of a directory or file
Group	Group of a directory or file
Permission	Permission of a directory or file
Date	Time when a directory or file is created

Step 3 In the search box, enter a keyword. The system automatically searches directories or files in the current directory.

Step 4 Clear the search criteria. The system displays all directories or files.

----End

Performing Actions

Step 1 On the **File Browser** page, select one or more directories or files.

Step 2 Click **Actions**. On the menu that is displayed, select an operation.

- **Rename**: renames a directory or file.
- **Move**: moves a file. In **Move to**, select a new directory and click **Move**.
- **Copy**: copies the selected files or directories.
- **Change permissions**: changes permission to access the selected directory or file.
  - You can grant the owner, the group, or other users with the **Read**, **Write**, and **Execute** permissions.
  - **Sticky**: indicates that only HDFS administrators, directory owners, and file owners can move files in the directory.
  - **Recursive**: indicates that permission is granted to subdirectories recursively.
- **Storage policies**: indicates the policies for storing files or directories in HDFS.
- **Summary**: indicates that the HDFS storage information about the selected file or directory can be viewed.

----End

Uploading User Files

Step 1 On the **File Browser** page, click **Upload**.

Step 2 In the displayed dialog box for uploading files, click **Select files** or drag the file to the dialog box.

----End

Creating a New File or Directory

Step 1 On the **File Browser** page, click **New**.

Step 2 Select an operation.

- **File**: creates a file. Enter a file name and click **Create**.
- **Directory**: creates a directory. Enter a directory name and click **Create**.

----End

Storage Policy Definition and Usage

NOTE

If the value of Hue parameter **fs_defaultFS** is set to **viewfs://ClusterX**, the big data storage policy cannot be enabled.

Step 1 Log in to FusionInsight Manager.

Step 2 On FusionInsight Manager, choose **System** > **Permission** > **Manage Role** > **Create Role**.

1. Set **Role Name**.
2. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > **Hue**, select **Storage Policy Admin**, and click **OK**. Then, grant the permission to the role.

Step 3 Choose **System** > **Permission** > **User Group** > **Create User Group**. Set **Group Name** and click **Select and Add Role** next to **Role**. On the displayed page, select the role created in **Step 2** and click **OK** to add the role to the group.

Step 4 Choose **System** > **Permission** > **User** > **Create**.

1. **Username**: Enter the name of the user to be added.
2. Set **User Type** to **Human-machine**.
3. Set **Password** and **Confirm Password** for logging in to the Hue web UI.
4. Click **Add** next to **User Group**. On the page that is displayed, select the user group created in **Step 3**, **supergroup**, **hadoop**, and **hive**, and click **OK**.
5. Set **Primary Group** to **hive**.
6. Click **Add** on the right of **Role**. On the page that is displayed, select the role created in **Step 2** and **System_administrator** role, and click **OK**.
7. Click **OK**. The user is added successfully.

Step 5 Access the Hue web UI as the created user. For details, see **Accessing the Hue Web UI**.

Step 6 In the left navigation tree, click  . The **File Browser** page is displayed.

Step 7 Select the check box of the directory and click **Actions** on the top of the page. Choose **Storage policies**.

Step 8 In the dialog box that is displayed, set a new storage policy and click **OK**.

----End

12.12.8 Using Job Browser on the Hue Web UI

Scenario

Users can use the Hue web UI to query all jobs in an MRS cluster.

Accessing Job Browser

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click  .

View the jobs in the current cluster.

NOTE

The number on **Job Browser** indicates the total number of jobs in the cluster.

Job Browser displays the following job information:

Table 12-246 MRS job attributes

Attribute	Description
Name	Job name
User	User who starts a job
Type	Job type
Status	Job status, including **Succeeded**, **Running**, and **Failed**.
Progress	Job running progress
Group	Group to which a job belongs
Start	Start time of a job
Duration	Job running duration
Id	Job ID, which is generated by the system automatically.

NOTE

If the MRS cluster has Spark, the **Spark-JDBCServer** job is started by default to execute tasks.

----End

Searching for Jobs

Step 1 In the search box of **Job Browser**, enter the specified character. The system automatically searches for all jobs that contain the keyword by ID, name, or user.

Step 2 Clear the search criteria. The system displays all jobs.

----End

Querying Job Details

Step 1 In the job list on the **Job Browser** page, click the row that contains the desired job to view details.

Step 2 On the **Metadata** tab page, you can view the metadata of the job.

NOTE

You can click **Log** to open the job running log.

----End

12.12.9 Using HBase on the Hue Web UI

Scenario

You can use Hue to create or query HBase tables in a cluster and run tasks on the Hue web UI.

Make sure that the HBase component has been installed in the MRS cluster and the Thrift1Server instance has been added before this operation.

Accessing Job Browser

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click HBase  . The **HBase Browser** page is displayed.

----End

Creating an HBase Table

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click HBase  . The **HBase Browser** page is displayed.

Step 3 Click **New Table** on the right, enter the table name and column family parameters, and click **Submit**.

----End

Querying Data in an HBase Table

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click HBase  . The **HBase Browser** page is displayed.

Step 3 Click the HBase table to be queried. Then, click the key value next to search box in the upper part, and query the HBase table.

----End

12.12.10 Typical Scenarios

12.12.10.1 HDFS on Hue

Hue provides the file browser function for users to use HDFS in GUI mode.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

How to Use File Browser

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Click  . The **File Browser** page is displayed. You can perform the following operations:

- Viewing files or directories

  By default, the directory and files in the directory of the login user are displayed. You can view **Name**, **Size**, **User**, **Group**,**Permission**, and **Date**.

  Click a file name to view the text information or binary data in the text file. The file content can be edited.

  If there are a large number of files and directories, you can enter keywords in the search text box to search for specific files or directories.

- Creating files or directories

  Click **New** in the upper right corner. Choose **File** to create the file. Choose **Directory** to create a directory.

- Managing files or directories

  Select the check box of a file or director, and click **Actions**. In the displayed menu, choose **Rename**, **Move**, **Copy**, and **Change permissions** to rename, move, copy, or change the file or directory permissions.

- Uploading files

  Click **Upload** in the upper right corner and click **Select files** or drag the file to the window.

How to Use Storage Policies

NOTE

If the value of Hue parameter **fs_defaultFS** is set to **viewfs://ClusterX**, the big data storage policy cannot be enabled.

Storage policies on the Hue web UI are classified into the following two types:

- Static Storage Policies

  Current storage policy

  According to the access frequency and importance of documents in HDFS, specify a storage policy for an HDFS directory, such as ONE_SSD or ALL_SSD. The files in this directory can be migrated to the storage media.

- Dynamic Storage Policies

  Set rules for an HDFS directory. The system can automatically change the storage policy, the number of file copies, migrate the file directory..

  Before configuring a dynamic storage policy on the Hue WebUI, you must set the CRON expressions for cold and hot data migration and start automatic cold and hot data migration on Manager.

  Operations:

  Change the value of **dfs.auto.data.mover.cron.expression** for NameNode of the HDFS service. For details, see **Modifying Cluster Service Configuration Parameters**.

  NOTE

  - **dfs.auto.data.mover.cron.expression** indicates the CRON expression for checking whether HDFS data meets the dynamic storage policy rule. It is used to control the start time of data migration. The default value is **0 * * * ***, indicating that the detection is performed on the hour. When the dynamic storage policy rule is met, cold and hot data migration tasks are executed on the hour.
  - The default value of **dfs.auto.data.mover.enable** is **false**. This parameter value is valid only when **dfs.auto.data.mover.enable** is set to **true**.

  Table 12-247 describes the expression for modifying this parameter. ***** indicates consecutive time segments.

  Table 12-247 Parameters in the execution expression

Column	Description
1	Minute. The value ranges from 0 to 59.
2	Hour. The value ranges from 0 to 23.
3	Date. The value ranges from 1 to 31.
4	Month. The value ranges from 1 to 12.
5	Week. The value ranges from 0 to 6. **0** indicates Sunday.

To set storage policies on the web UI, perform the following operations:

Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 2 On FusionInsight Manager, choose **System** > **Permission > Role** > **Create Role**.

1. Set **Role Name**.
2. In the **Configure Resource Permission** area, choose *Name of the desired cluster* > **Hue**, select **Storage Policy Admin**, and click **OK**. Then, grant the permission to the role.

Step 3 Choose **System** > **Permission** > **User Group** > **Create User Group**. Set **Group Name**, and click **Add** next to **Role**. On the displayed page, select the created role, click **OK** to add the role to the group, and click **OK**.

Step 4 Choose **System** > **Permission** > **User** > **Create**.

1. **Username**: Enter the name of the user to be added.
2. Set **User Type** to **Human-machine**.
3. Set **Password** and **Confirm Password** for logging in to the Hue web UI.
4. Click **Add** next to **User Group**. On the page that is displayed, select the created user group in **Step 3**, **supergroup**, **hadoop**, and **hive**, and click **OK**.
5. Set **Primary Group** to **hive**.
6. Click **Add** next to **Role**. On the page that is displayed, select the created role in **Step 2** and the **System_administrator** role, and click **OK**.
7. Click **OK**. The user is added successfully.

Step 5 Access the Hue web UI as the created user. For details, see **Accessing the Hue Web UI**.

Step 6 In the left navigation pane, click  . The **File Browser** page is displayed.

Step 7 Select the check box of a directory and choose **Action** on the top of the page. Choose **Storage policies**.

Step 8 In the dialog box that is displayed, set a new storage policy and click **OK**.

- On the **Static Storage Policy** page, you can set a static storage policy and click **Save**.
- On the **Dynamic Storage Policy** page, you can create, delete, or modify a dynamic storage policy. **Table 12-248** describes the parameters.

Table 12-248 Parameters of the dynamic storage policy

Category	Parameter	Description
Rule	Last Access to File	Indicates the time when the file is last accessed.
	Last File Modification	Indicates the time when the file is last modified.
Operation	Change Number of Copies	Indicates the number of file copies.

Category	Parameter	Description
	Modify Storage Policy	Indicates that you can modify storage policies to the following: HOT, WARM, COLD, ONE_SSD, and ALL_SSD.
	Move to Directory	Indicates that you can move the file to another directory.

NOTE

- You need to consider whether the rules conflict with each other and whether the rules damage the system when setting rules.
- When a directory is configured with multiple rules and operations, the rule that is triggered first is located at the bottom of the rule/operation list, and the rules that are triggered later are placed from bottom to top to prevent repeated operations.
- The system checks whether the files under the directory specified by the dynamic storage policy meet the rules on an hourly basis. If the files meet the rules, the execution is triggered. Execution logs are recorded in the **/var/log/Bigdata/hdfs/nn/hadoop.log** directory of the active NameNode.

----End

Typical Scenarios

On the Hue page, view and edit HDFS files in text or binary mode as follows:

Viewing a File

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the left navigation pane, click  . The **File Browser** page is displayed.

Step 3 Click the name of the file to be viewed.

Step 4 Click **View as binary** to switch from the text mode to the binary mode. Click **View as file** to switch from the binary mode to the text mode.

Editing a file

Step 5 Click **Edit File**. The file content can be edited.

Step 6 Click **Save** or **Save As** to save the file.

----End

12.12.10.2 Configuring HDFS Cold and Hot Data Migration

Scenario

The hot and cold data migration tool migrates HDFS files based on the configured policy. A policy is a set of conditional or non-conditional rules. If a file matches the rule set, the tool performs a group of operations for the file.

The hot and cold data migration tool supports the following rules and operations:

- Migration rules:
  - Data is migrated based on the latest access time of the file.
  - Data is migrated based on the file modification time.
  - Data is migrated without conditions.

  Table 12-249 Rule condition tags

Condition Tag	Description
<age operator="lt">	Defines the conditions for changing the age or modification time.
<atime operator="gt">	Defines the condition for accessing time.

  NOTE

  For a manual migration rule, no condition is required.

- Operations:
  - Set the storage policy to a given data tier.
  - Migrate files to another folder.
  - Configure the number of copies for a file.
  - Delete a file.
  - Set a node label.

  Table 12-250 Behavior types:

Behavior Type	Description	Required Parameters
MARK	Determines the data access frequency and set a data storage policy.	<param> <name>targettier</name> <value>STORAGE_POLICY</value> <param>

Behavior Type	Description	Required Parameters
MOVE	Sets the data storage policy or NodeLabel and invokes the HDFS Mover tool.	<param> <name>targettier</name> <value>STORAGE_POLICY</value> <param> <param> <name>targetnodelabels</name> <value>SOME_EXPRESSION</value> <param> **NOTE** You can set either or both of the parameters.
SET_REPL	Configures the number of copies for a file.	<param> <name>replcount</name> <value>INTEGER</value> <param>
MOVE_TO_FOLDER	Moves the file to the target folder. If **overwrite** is set to **true**, the target path will be overwritten.	<param> <name>target</name> <value>PATH</value> <param> <param> <name>overwrite</name> <value>true/false</value> <param> **NOTE** **overwrite** is an optional parameter. If this parameter is not set, the default value **false** is used.
DELETE	Delete a file.	N/A

Configuration Description

You must periodically invoke the migration tool and perform the following operations in the **hdfs-site.xml** file on the client:

Table 12-251 Parameter description

Parameter	Description	Default Value
dfs.auto-data-movement.policy.class	Specifies the default data migration policy. **NOTE** Currently, only **DefaultDataMovementPolicy** is supported.	com.xxx.hadoop.hdfs.datamovement.policy.DefaultDataMovementPolicy
dfs.auto.data.mover.id	Specifies the output file name of the hot and cold data migration policy.	Current system time (ms)
dfs.auto.data.mover.output.dir	Specifies the name of the HDFS directory to which cold and hot data is migrated. The migration tool writes the behavior status file here.	/system/datamovement

DefaultDataMovementPolicy has the configuration file **default-datamovement-policy.xml**. Users need to define all rules based on the age or access time and operations performed in this file. This file must be stored in **classpath** of the client.

The following is an example of the **default-datamovement-policy.xml** file:

```
<policies>
  <policy>
    <fileset>
      <file>
        <name>/opt/data/1.txt</name>
      </file>
      <file>
        <name>/opt/data/*/subpath/</name>
        <excludes>
          <name>/opt/data/some/subpath/sub1</name>
        </excludes>
      </file>
    </fileset>
    <rules>
      <rule>
        <age>2w</age>
        <action>
          <type>MOVE</type>
          <params>
            <param>
              <name>targettier</name>
              <value>HOT</value>
            </param>
          </params>
        </action>
      </rule>
    </rules>
  </policy>
</policies>
```

NOTE

Other attributes can be added to the tags used in policies, rules, and behavior operations. For example, **name** can be used to manage the mapping between the user UI (for example, Hue UI) and tool input XML.

Example: **<policy name="Manage_File1">**

The tags are described as follows:

Table 12-252 Description of configuring tags

Tag	Description	Reusable or Not
<policy>	Define a single policy. • **idempotent**: specifies whether to check the next rule if the current rule is met when multiple rules exist in the policy. Example: **<policy name ="policy2" idempotent ="true">** The default value is **true**, indicating that the rule and action are idempotent and you can continue to check the next rule. If the value is **false**, the evaluation stops at the current rule. • **hours_allowed**: indicates whether to execute policy evaluation based on the system time. The value of **hours_allowed** is a number separated by commas (,). The value ranges from 0 to 23, indicating the system time. Example: **<policy name ="policy1" hours_allowed ="2-6,13-14">** If the current system time is within the configured range, continue the evaluation. Otherwise, the evaluation will be skipped. **NOTE** In the input XML, only one policy is supported per file. Therefore, all rules in the file must be covered by a policy tag.	Yes
<fileset>	Define a group of files or folders for each policy.	No (in the **policy** tag)
<file>	One or more **<name>** tags are configured for the definition file and/or folder in the **<file>** tag. The file or folder name supports POSIX globs.	Yes (in the **fileset** tag)
<excludes>	Define this tag in the **<file>** tag. This tag can contain multiple **<name>** tags. In the file or folder range configured in the **<file>** tag, the files or folders contained in the **<name>** tag will be excluded. The file or folder name supports POSIX globs.	No (in the **fileset** tag)

Tag	Description	Reusable or Not
<rules>	Specifies multiple rules defined for a policy.	No (in the **policy** tag)
<rule>	Specifies a single rule to be defined.	Yes (in the **rules** tag)
<age>or<atime>	Defines the age/accesstime of the file defined in **<fileset>**. The policy matches the age. The value of **age** can be in the *[num]y[num]m[num]w[num]d[num]h* format. In the command, *num* indicates a number. The meanings of the letters are as follows: * *y*: year (365 days in a year) * *m*: month (30 days in a month) * *w*: week (7 days in a week) * *d*: day * *h*: hour You can use the year, month, week, day, or hour independently, or you can combine them. For example, **1y2d** indicates one year and two days or 367 days. If there is no unit (that is, the number is not followed by any letter), the default unit is day. **NOTE** You can configure **gt** (greater) and **lt** (less) in the **<age>** and **<atime>** tags. The default operator is **gt**. Example: **<age operator="lt">**	No (in the **rule** tag)
<action>	If the rule is matched, this tag defines the action to be executed.	No (in the **rule** tag)
<type>	Defines the action type. Currently, the supported action types are MOVE and MARK.	No (in the **action** tag)
<**params**>	Defines parameters related to each action.	No (in the **action** tag)

Tag	Description	Reusable or Not
<param>	Defines a name-value format parameter that uses the **<name>** and **<value>** tags. For MARK and MOVE, only the **targettier** parameter is supported. This parameter specifies the data storage policy if the age rule is met. If multiple parameters have the same name, the first parameter value is used. For marks, the supported **targettier** values are **ALL_SSD**, **ONE_SSD**, **HOT**, **WARM**, and **COLD**. For MOVE, the supported **targettier** values are **ALL_SSD**, **ONE_SSD**, **HOT**, **WARM**, and **COLD**.	Yes (in the **params** tag)

For files or folders under the **<file>** tag, the **FileSystem#globStatus** API is used. For other files or folders, the **GlobPattern** class (used by GlobFilter) is used. For details, see the description of supported APIs. For example, for globStatus, **/opt/hadoop/*** will match everything in the **/opt/hadoop** folder. **/opt/*/hadoop** matches all hadoop folders in the subdirectories of the**/opt** directory.

For globStatus, the glob mode of each path component is matched. For other components, the glob mode is directly matched.

https://hadoop.apache.org/docs/r3.1.1/api/org/apache/hadoop/fs/FileSystem.html#globStatus(org.apache.hadoop.fs.Path)

Glob	Name	Matches
*	*asterisk*	Matches zero or more characters
?	*question mark*	Matches a single character
[ab]	*character class*	Matches a single character in the set {a, b}
[^ab]	*negated character class*	Matches a single character that is not in the set {a, b}
[a-b]	*character range*	Matches a single character in the (closed) range [a, b], where a is lexicographically less than or equal to b
[^a-b]	*negated character range*	Matches a single character that is not in the (closed) range [a, b], where a is lexicographically less than or equal to b
{a,b}	*alternation*	Matches either expression a or b
\c	*escaped character*	Matches character c when it is a metacharacter

Behavior Operation Example

- MARK

```
<action>
  <type>MARK</type>
  <params>
    <param>
      <name>targettier</name>
      <value>HOT</value>
    </param>
  </params>
</action>
```

- MOVE

```
<action>
 <type>MOVE</type>
 <params>
  <param>
   <name>targettier</name>
   <value>HOT</value>
  </param>
  <param>
   <name>targetnodelabels</name>
   <value>SOME_EXPRESSION</value>
  </param>
 </params>
</action>
```

- SET_REPL

```
<action>
 <type>SET_REPL</type>
 <params>
  <param>
   <name>replcount</name>
   <value>5</value>
  </param>
 </params>
</action>
```

- MOVE_TO_FOLDER

```
<action>
 <type>MOVE_TO_FOLDER</type>
 <params>
  <param>
   <name>target</name>
   <value>path</value>
  </param>
  <param>
   <name>overwrite</name>
   <value>true</value>
  </param>

 </params>
</action>
```

NOTE

The **MOVE_TO_FOLDER** operation only changes the file path to the target folder and does not change the block location. If you want to move a block, you need to configure an independent move policy.

- DELETE

```
<action>
 <type>DELETE</type>
</action>
```

NOTE

- When writing an XML file, pay attention to the configuration and sequence of behavior operations. The hot and cold data migration tool executes the rules in the sequence specified in the input XML file.
- If you want to run only one rule based on **atime/age**, sort the rules in descending order of time and set the idempotent attribute to false.
- If the delete operation is configured for a file set, other rules cannot be configured after the delete operation is performed.
- The **-fs** option can be used to specify the default file system address of the client.

Audit Logs

The cold and hot data migration tool supports audit logs of the following operations:

- Tool startup status
- Behavior type, parameter details, and status
- Tool completion status

To enable the audit log tool, add the following attributes to the **<HADOOP_CONF_DIR>/log4j.property** file:

```
autodatatool.logger=INFO, ADMTRFA
autodatatool.log.file=HDFSAutoDataMovementTool.audit
log4j.logger.com.xxx.hadoop.hdfs.datamovement.HDFSAutoDataMovementTool.audit=${autodatatool.logger}
log4j.additivity.com.xxx.hadoop.hdfs.datamovement.HDFSAutoDataMovementTool-audit=false
log4j.appender.ADMTRFA=org.apache.log4j.RollingFileAppender
log4j.appender.ADMTRFA.File=${hadoop.log.dir}/${autodatatool.log.file}
log4j.appender.ADMTRFA.layout=org.apache.log4j.PatternLayout
log4j.appender.ADMTRFA.layout.ConversionPattern=%d{ISO8601} %p %c: %m%n
log4j.appender.ADMTRFA.MaxBackupIndex=10
log4j.appender.ADMTRFA.MaxFileSize=64MB
```

NOTE

For details, see the **<HADOOP_CONF_DIR>/log4j_autodata_movment_template.properties** file.

12.12.10.3 Hive on Hue

Hue provides the Hive GUI management function so that users can query Hive data in GUI mode.

How to Use Query Editor

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

In the navigation tree on the left, click  and choose **Hive**. The **Hive** page is displayed.

- Running Hive HQL statements

  Select the target database on the left. You can also click  in the upper right corner and enter the target database name to search for the target database.

  Enter a Hive HQL statement in the text box and click  or press **Ctrl+Enter** to run the HQL statement. The execution result is displayed on the **Result** tab page.

- Analyzing Hive HQL statements

  Select the target database on the left, enter the Hive HQL statement in the text box, and click  to compile the HQL statement and check whether the statement is correct. The execution result is displayed under the text editing box.

- Saving HQL statements

Enter the Hive HQL statement in the text box, click in the upper right corner, and enter the name and description. You can view the saved statements on the **Saved Queries** tab page.

- Viewing historical records

  Click **Query History** to view the HQL running status. You can view the history of all the statements or only the saved statements. If many historical records exist, you can enter keywords in the text box to search for desired records.

- Configuring advanced query

  Click in the upper right corner to configure the file, function, and settings.

- Viewing the information of shortcut keys

  Click in the upper right corner to view information about all shortcut keys.

How to Use Metadata Browser

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

- Viewing metadata of Hive tables

  Click in the navigation tree on the left and click a table name. The metadata of the Hive table is displayed.

- Managing metadata of Hive tables

  On the metadata information page of a Hive table:

  - Click **Import** in the upper right corner to import data.
  - Click **Overview** to view the location of the table file in the **PROPERTIES** field.

    View the field information of each column in a Hive table and manually add description information. Note that the added description information is not the field comments in the Hive table.
  - Click **Sample** to browse data.

- Managing Hive metadata tables

  Click in the left list to create a table based on the uploaded file in the database. You can also manually create a table.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

Typical Scenarios

On the Hue page, create a Hive table as follows:

Step 1 Click  at the upper left corner of Hue web UI and select the Hive instance to be operated to enter the Hive command execution page.

Step 2 Enter an HQL statement in the command input box, for example:

create table hue_table(id int,name string,company string) row format delimited fields terminated by ',' stored as textfile;

Click  to execute the HQL statements.

Step 3 Enter the following command in the command input box:

show tables;

Click  to view the created table **hue_table** in **Result**.

----End

12.12.10.4 Oozie on Hue

Hue provides the Oozie job manager function, in this case, you can use Oozie in GUI mode.

CAUTION

The Hue page is used to view and analyze data such as files and tables. Do not perform high-risk management operations such as deleting objects on the page. If an operation is required, you are advised to perform the operation on each component after confirming that the operation has no impact on services. For example, you can use the HDFS client to perform operations on HDFS files and use the Hive client to perform operations on Hive tables.

How to Use Oozie Job Designer

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

In the navigation tree on the left, click  and choose **Workflow**.

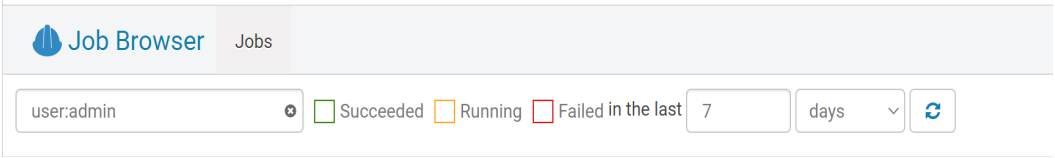
The job designer allows users to create MapReduce, Java, Streaming, Fs, SSH, Shell and DistCp jobs.

How to Use Dashboard

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Click **Jobs** in the upper right corner. The **Job Browser** page is displayed.

View the running status of the Workflow, Coordinator, and Bundles jobs.



How to Use Editor

Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

In the navigation tree on the left, click  and choose **Workflow**.

Workflows, Schedule, and Bundle tasks can be created. Existing applications can be submitted for running, shared, copied, and exported.

- Each Workflow can contain one or more jobs to form a complete workflow for a specified service.

  When creating a Workflow, you can design jobs in the Hue editor and add the jobs to the Workflow.
- Each Schedule can define a time trigger to periodically execute a specified Workflow. One time trigger cannot execute multiple Workflows.
- Each Bundles can define a set to execute multiple Schedules so that different Workflows can be executed in batches.

12.12.11 Hue Log Overview

Log Description

Log paths: The default paths of Hue logs are **/var/log/Bigdata/hue** (for storing run logs) and **/var/log/Bigdata/audit/hue** (for storing audit logs).

Log archive rules: The automatic compression and archiving function of the Hue logs is enabled. By default, when the size of a log file (**access.log**, **error.log**, **runcpserver.log**, or **hue-audits.log**) exceeds 5 MB, logs are automatically compressed. A maximum of 20 latest compressed files are reserved. The number of compressed files and compression threshold can be configured.

Table 12-253 Hue log list

Type	Log File Name	Description
Run log	access.log	Access log file
	error.log	Error log file
	gsdb_check.log	Log file of the GaussDB check information
	kt_renewer.log	Log file of Kerberos authentication
	kt_renewer.out.log	Log file of the abnormal Kerberos authentication logs
	runcpserver.log	Log file of operation records
	runcpserver.out.log	Log file of process running exceptions
	supervisor.log	Log file of process startup

Type	Log File Name	Description
	supervisor.out.log	Log file of process startup exceptions
	dbDetail.log	Log file of database initialization
	initSecurityDetail.log	Download initialization log file of the Keytab file
	postinstallDetail.log	Work log file generated after the Hue service is installed
	prestartDetail.log	Prestart log file
	statusDetail.log	Log file of the Hue health status
	startDetail.log	Startup log
	get-hue-ha.log	Log file of the Hue HA status
	hue-ha-status.log	Log file of the Hue HA status monitoring
	get-hue-health.log	Log file of the Hue health status
	hue-health-check.log	Log file of the Hue health check
	hue-refresh-config.log	Log file of the Hue configuration update
	hue-script-log.log	Log file of the Hue operations on the Manager console
	hue-service-check.log	Log file of the Hue service status monitoring
	db_pwd.log	Log that records the changes of the password for Hue to connect to the DBService database
	modifyDBPwd_*Date*.log	-
	watch_config_update.log	Parameter update log file
Audit log	hue-audits.log	Audit log file

Log Level

Table 12-254 describes the log levels supported by Hue.

Levels of logs are ERROR, WARN, INFO, and DEBUG from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-254 Log levels

Level	Description
ERROR	Logs of this level record error information about system running.
WARN	Logs of this level record exception information about the current event processing.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of the Hue service by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 In the navigation tree on the left, select **Log** corresponding to the role to be modified.

Step 3 Select the log level to be changed on the right.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

Step 5 Restart the service or instance whose configuration has expired for the configuration to take effect.

----End

Log Format

The following table lists the Hue log formats:

Table 12-255 Log formats

Type	Format	Example
Run log	*<dd-MM-yy HH:mm:ss,SSS><Location where the log event occurs><Log level><Message in the log>*	[03/Nov/2014 11:57:19] middleware | INFO | Unloading MimeTypeJSFileFixStrea-mingMiddleware.

Type	Format	Example
	<Log level><Time format><yyyy-MM-dd HH:mm:ss,SSS><Location where the log event occurs><Message in the log>	INFO : CST 2014-11-06 11:22:52 hue-ha-status.sh : update 4 <= 15:myHostName=10.0.0.250 ACTIVE=10.0.0.250
Audit log	*<UserName><yyyy-MM-dd HH:mm:ss,SSS>< Audit operation description> <Resource parameter> <URL> <Whether to allow> <Audit operation> <IP address>*	{"username": "admin", "eventTime": "2014-11-06 10:28:34", "operationText": "Successful login for user: admin", "service": "accounts", "url": "/accounts/login/", "allowed": true, "operation": "USER_LOGIN", "ipAddress": "10.0.0.250"}

12.12.12 Common Issues About Hue

12.12.12.1 How Do I Solve the Problem that HQL Fails to Be Executed in Hue Using Internet Explorer?

Question

What do I do if all HQL statements fail to be executed when I use Internet Explorer to access Hive Editor in Hue and the message "There was an error with your query" is displayed?

Answer

Internet Explorer does not support processing of AJAX POST requests containing form data in 307 redirection. You are advised to use a compatible browser, for example, Google Chrome.

12.12.12.2 Why Does the use database Statement Become Invalid When Hive Is Used?

Question

When Hive is used, the **use database** statement is entered in the text box to switch the database, and other statements are also entered, why does the database fail to be switched?

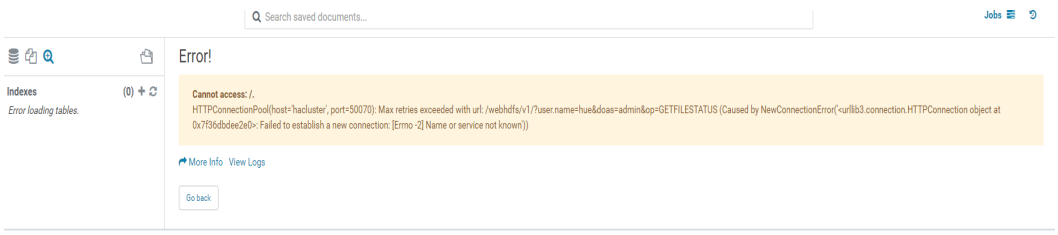
Answer

Using Hive on Hue is different from using Hive on the Hive client. There is an option to select a database on the Hue interface, and the database where the current SQL is executed is the one that is displayed on the interface. You are advised to use functions on the Hue interface instead of using statements to perform session-level and one-off operations, for example, setting parameters. If you must enter specific statements to perform an operation, ensure that all statements you enter are in one text box.

12.12.12.3 What Can I Do If HDFS Files Fail to Be Accessed Using Hue WebUI?

Question

What can I do if an error message shown in the following figure is displayed, indicating that the HDFS file cannot be accessed when I use Hue web UI to access the HDFS file?



Answer

1. Check whether the user who logs in to the Hue web UI has the permissions of the **hadoop** user group.
2. Check whether the HttpFS instance has been installed for the HDFS service and is running properly. If the HttpFS instance is not installed, manually install and restart the Hue service.

12.12.12.4 What Can I Do If a Large File Fails to Be Uploaded on the Hue Page?

Question

What can I do when a large file fails to be uploaded on the Hue page?

Answer

1. You are advised to run commands on the client to upload large files instead of using the Hue file browser.
2. If you must use Hue to upload the file, perform the following steps to modify Httpd parameters:
   a. Log in to the active management node as user **omm**.
   b. Run the following command to edit the **httpd.conf** file:
      vi $BIGDATA_HOME/om-server/Apache-httpd-*/conf/httpd.conf

c. Search for **21201** and add **RequestReadTimeout handshake=0 header=0 body=0** to the **</VirtualHost>** configuration, as shown in the following:

```
...
<VirtualHost *:21201>
   ServerName https://10.112.16.93:21201
   AllowEncodedSlashes On
   SSLProxyEngine On
   ProxyRequests Off
   TraceEnable off
   ProxyTimeout  1200
   RewriteEngine on
   RewriteMap proxylist dbm:${BIGDATA_ROOT_HOME}/om-server_*/Apache-httpd-*/conf/
proxylist.dbm

   RewriteRule ^(\/.*)$  ${proxylist:/Hue/Hue/21201}$1 [E=TARGET_PATH:$1,L,P]

   Header edit Location ^(?!https://10.112.16.93:20009|https://
10.112.16.93:21201)http[s]?://[^/]*(.*)$  https://10.112.16.93:21201$1

   ProxyPassReverseCookiePath / / interpolate

   SSLEngine On
   SSLProxyProtocol  All +TLSv1.2 -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
   SSLProtocol ALL +TLSv1.2 -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
   SSLCipherSuite ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-
SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:DHE-DSS-
AES256-GCM-SHA384:DHE-RSA-AES256-GCM-SHA384:DHE-DSS-AES128-GCM-SHA256:DHE-
RSA-AES128-GCM-SHA256
   SSLProxyCheckPeerName off
   SSLProxyCheckPeerCN off
   SSLCertificateFile "${BIGDATA_ROOT_HOME}/om-server_*/Apache-httpd-*/conf/security/
proxy_ssl.cert"
   SSLCertificateKeyFile "${BIGDATA_ROOT_HOME}/om-server_*/Apache-httpd-*/conf/security/
server.key"
   SSLProxyCACertificateFile ${BIGDATA_ROOT_HOME}/om-server_*/apache-tomcat-*/conf/
security/tomcat.crt
   SSLCertificateChainFile "${BIGDATA_ROOT_HOME}/om-server_*/Apache-httpd-2.4.39/conf/
security/proxy_chain.cert"
   RequestReadTimeout handshake=0 header=0 body=0
</VirtualHost>
...
```

d. Run the **ps -ef|grep httpd|grep -v grep|xargs kill -9** command to restart **httpd**.

12.12.12.5 Why Is the Hue Native Page Cannot Be Properly Displayed If the Hive Service Is Not Installed in a Cluster?

Question

Why is the native Hue page blank if the Hive service is not installed in a cluster?

Answer

In MRS 3.x, Hue depends on Hive. If this problem occurs, check whether the Hive component is installed in the current cluster. If not, install it.

12.13 Using Impala

12.13.1 Using Impala from Scratch

Impala is a massively parallel processing (MPP) SQL query engine for processing vast amounts of data stored in Hadoop clusters. It is an open source software written in C++ and Java. It provides high performance and low latency compared with other SQL engines for Hadoop.

Background

Suppose a user develops an application to manage users who use service A in an enterprise. The procedure of operating service A on the Impala client is as follows:

Operations on common tables:

- Create the **user_info** table.
- Add users' educational backgrounds and titles to the table.
- Query user names and addresses by user ID.
- Delete the user information table after service A ends.

Table 12-256 User information

No.	Name	Gender	Age	Address
12005000201	A	Male	19	City A
12005000202	B	Female	23	City B
12005000203	C	Male	26	City C
12005000204	D	Male	18	City D
12005000205	E	Female	21	City E
12005000206	F	Male	32	City F
12005000207	G	Female	29	City G
12005000208	H	Female	30	City H
12005000209	I	Male	26	City I
12005000210	J	Female	25	City J

Prerequisites

The client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the Impala client command to implement service A.

Run the client command of the Impala component directly.

impala-shell

NOTE

By default, **impala-shell** attempts to connect to the Impala daemon on port 21000 of **localhost**. To connect to another host, use the **-i < host:port >** option, for example, **impala-shell -i** *xxx.xxx.xxx.xxx***:21000**. To automatically connect to a specific Impala database, use the **-d <database>** option. For example, if all your Kudu tables are in the **impala_kudu** database, **-d impala_kudu** can use this database. To exit the Impala Shell, run the **quit** command.

Operations on internal tables:

1. Create the **user_info** user information table according to **Table 12-256** and add data to it.
   ```
   create table user_info(id string,name string,gender string,age int,addr string);
   insert into table user_info(id,name,gender,age,addr) values("12005000201", "A", "Male", 19, "City A");
   ```
   ... (Other statements are the same.)
2. Add users' educational backgrounds and titles to the **user_info** table.

   For example, to add educational background and title information about user 12005000201, run the following commands.
   ```
   alter table user_info add columns(education string,technical string);
   ```
3. Query user names and addresses by user ID.

   For example, to query the name and address of user 12005000201, run the following command:
   ```
   select name,addr from user_info where id='12005000201';
   ```
4. Delete the user information table:
   ```
   drop table user_info;
   ```

Operations on external partition tables:

Create an external partition table and import data.

1. Create a path for storing external table data.
   - Security mode (Kerberos authentication is enabled for clusters)

     cd /opt/hadoopclient

     source bigdata_env

     kinit *hive*

     NOTE

     The user must have the hive administrator permissions.

     impala-shell

     hdfs dfs -mkdir /hive

     hdfs dfs -mkdir /hive/user_info

- Normal mode (Kerberos authentication is disabled for clusters)

  su - omm

  cd /opt/hadoopclient

  source bigdata_env

  impala-shell

  hdfs dfs -mkdir /hive

  hdfs dfs -mkdir /hive/user_info

2. Create a table.

```
create external table user_info(id string,name string,gender string,age int,addr string) partitioned by(year string) row format delimited fields terminated by ' ' lines terminated by '\n' stored as textfile location '/hive/user_info';
```

   NOTE

   fields terminated indicates delimiters, for example, spaces.

   lines terminated indicates line breaks, for example, **\n**.

   /hive/user_info indicates the path of the data file.

3. Import data.

   a. Execute the **insert** statement to insert data.

```
insert into user_info partition(year="2018") values ("12005000201", "A", "Male", 19, "City A");
```

   b. Run the **load data** command to import file data.

      i. Create a file based on the data in **Table 12-256**. For example, the file name is **txt.log**. Fields are separated by space, and the line feed characters are used as the line breaks.

      ii. Upload the file to HDFS.

      hdfs dfs -put txt.log /tmp

      iii. Load data to the table.

      load data inpath '/tmp/txt.log' into table user_info partition (year='2018');

4. Query the imported data:

```
select * from user_info;
```

5. Delete the user information table:

```
drop table user_info;
```

----End

12.13.2 Accessing the Impala Web UI

You can view Impala job information on the Impala web UI. Impala web UIs are classified into the following types based on instances:

- **StateStore WebUI**: used to manage nodes.
- **Catalog WebUI**: used to view metadata.
- **Impalad WebUI**: used to view details about each SQL statement.

Prerequisites

Impala has been installed in a cluster.

Accessing the StateStore Web UI

Step 1 Access Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 2 Choose **Services** > **Impala**.

Step 3 In **StateStore WebUI** of **Impala Summary**, click **StateStore(Statestore)**. The StateStore web UI is displayed.

----End

Accessing the Catalog Web UI

Step 1 Access Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 2 Choose **Services** > **Impala**.

Step 3 In **Catalog WebUI** of **Impala Summary**, click **Catalog(Catalog)**. The Catalog web UI is displayed.

----End

Accessing the Impalad Web UI

Step 1 Access Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 2 Choose **Services** > **Impala** > **Instance**.

Step 3 Move the cursor to the Impalad instance in the **Role** column. The following link is displayed in the lower left corner of the page. Obtain the value after **null**, for example, **82** in this example.

https://*EIP*:9022/mrsmanager/index.jsp?locale=zh-cn#/app/services/Impala/Impalad/null/82/*EIP*/STARTED/status/detail

In the preceding command, **82** is an example. Change it based on the site requirements.

Step 4 For details, see **Accessing the StateStore Web UI**.

Step 5 Change **StateStore/*xx*** in the URL of the StateStore web UI to **Impalad/*xx*** and access the new URL, where *xx* is the value obtained in **Step 3**.

----End

12.13.3 Using Impala to Operate Kudu

You can use the SQL statements of Impala to insert, query, update, and delete data in Kudu as an alternative to using Kudu APIs to build custom Kudu applications.

Prerequisite

A complete cluster client has been installed. For example, the installation directory is **/opt/Bigdata/client**. The client directory in the following operations is only an example. Replace it with the actual installation directory.

Impala on Kudu

Step 1 Log in to the node where the client is installed.

Step 2 Run the following command to initialize environment variables:

source /opt/Bigdata/client/bigdata_env

Step 3 If Kerberos authentication is enabled for the cluster, perform the following operation to authenticate the user. If Kerberos authentication is not enabled for the cluster, skip this step.

kinit *Service user*

Step 4 Run the following command to log in to the Impala client:

impala-shell

NOTE

By default, **impala-shell** attempts to connect to the Impala daemon on port 21000 of **localhost**. To connect to another host, use the **-i <host:port>** option. To automatically connect to a specific Impala database, use the **-d <database>** option. For example, if all your Kudu tables are in the **impala_kudu** database, **-d impala_kudu** can use this database. To exit the Impala shell, run the **quit** command.

Step 5 Run the following commands to create an Impala table and import the prepared data, for example, data in the **/tmp/data10** directory:

create table dataorigin (name string,age string,pt string, date_p date) row format delimited fields terminated by ',' stored as textfile;

load data inpath '/tmp/data10' overwrite into table dataorigin;

Step 6 Run the following command to create a Kudu table. In the command, **kudu.master_addresses** indicates the IP address of the KuduMaster instance. Set it to the actual IP address.

create table dataorigin2 (name string,age string,pt string, date_p date, primary key(name)) stored as kudu TBLPROPERTIES('kudu.master_addresses'='192.168.190.164:7051,192.168.204.178:7051,192.168.244.63:7051');

Step 7 Perform the following operations on the Kudu table.

1. Insert data.

   insert into dataorigin2 select * from dataorigin;

2. Update data.

   UPDATE dataorigin2 SET date_p="2021-03-31" where age="73";

3. Upsert rows.

   UPSERT INTO dataorigin2 VALUES ("spjted","75","28","2021-03-32");

   UPSERT INTO dataorigin2 VALUES ("kwhakb","92","29","2021-03-33");

UPSERT INTO dataorigin2 VALUES ("oftrkf","13","30","2021-03-34");
UPSERT INTO dataorigin2 VALUES ("kiewti","36","31","2021-03-35");
UPSERT INTO dataorigin2 VALUES ("rknmql","98","32","2021-03-36");
UPSERT INTO dataorigin2 VALUES ("fwcoij","52","33","2021-03-37");
UPSERT INTO dataorigin2 VALUES ("pgvpdo","37","34","2021-03-35");

4. Delete a row.
   DELETE FROM dataorigin2 WHERE date_p="2021-03-31";

----End

12.13.4 Interconnecting Impala with External LDAP

This section applies to MRS 3.1.0 or later.

Step 1 Log in to Manager.

Step 2 On Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Impala** > **Configurations** > **All Configurations** > **Impalad(Role)** > **LDAP**.

Step 3 Set the following parameters.

Table 12-257 Parameter configuration

Parameter	Description	Remarks
--enable_ldap_auth	Whether to enable LDAP authentication	Value: **true** or **false**
--ldap_bind_pattern	LDAP user DN pattern	Example: **cn=#UID,ou=People,dc=xxx,dc=com** or **cn=%s,ou=People,dc=xxx,dc=com**

Parameter	Description	Remarks
--ldap_passwords_in_clear_ok	Whether the LDPA password is sent in plaintext	If this parameter is set to **true**, the LDAP password can be sent in plaintext. Value: **true** or **false** **NOTE** If **--enable_ldap_auth** is set to **true**, the LDAP TLS protocol is disabled by default during authentication. Therefore, you need to set **--ldap_passwords_in_clear_ok** to **true**. Otherwise, the Impalad role will fail to be started. To enable the Ldap TLS protocol, set **--ldap_tls** to **true** in the customized configuration of the Impalad role. After the configuration, the password can be sent in ciphertext.
--ldap_uri-ip	LDAP IP address	-
--ldap_uri-port	LDAP port number	Default value: **389**

Step 4 After the modification, click **Save** in the upper left corner. In the displayed dialog box, click **OK**.

Step 5 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Impala** > **Instance**. On the displayed page, select the instances whose **Configuration Status** is **Expired**, choose **More** > **Restart Instance**, and restart the instance.

----End

12.14 Using Kafka

12.14.1 Using Kafka from Scratch

Scenario

You can create, query, and delete topics on a cluster client.

Prerequisites

The client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Using the Kafka Client (Versions Earlier Than MRS 3.x)

Step 1 Access the ZooKeeper instance page.

Click the cluster name to go to the cluster details page and choose **Components** > **ZooKeeper** > **Instances**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

Step 2 View the IP addresses of the ZooKeeper role instance.

Record any IP address of the ZooKeeper instance.

Step 3 Log in to the node where the client is installed.

Step 4 Run the following command to switch to the client directory, for example, **/opt/hadoopclient/Kafka/kafka/bin**.

cd /opt/hadoopclient/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/hadoopclient/bigdata_env

Step 6 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Kafka user*

Step 7 Create a topic.

sh kafka-topics.sh --create --topic *Topic name* **--partitions** *Number of partitions occupied by the topic* **--replication-factor** *Number of replicas of the topic* **--zookeeper** *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

Step 8 Run the following command to view the topic information in the cluster:

sh kafka-topics.sh --list --zookeeper *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

Step 9 Delete the topic created in **Step 7**.

sh kafka-topics.sh --delete --topic *Topic name* **--zookeeper** *IP address of the node where the ZooKeeper instance resides*:*clientPort*/**kafka**

Type **y** and press **Enter**.

----End

Using the Kafka Client (MRS 3.x or Later)

Step 1 Access the ZooKeeper instance page.

Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Instance**.

Step 2 View the IP addresses of the ZooKeeper role instance.

Record any IP address of the ZooKeeper instance.

Step 3 Log in to the node where the client is installed.

Step 4 Run the following command to switch to the client directory, for example, **/opt/hadoopclient/Kafka/kafka/bin**.

cd /opt/hadoopclient/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/hadoopclient/bigdata_env

Step 6 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *Kafka user*

Step 7 Log in to FusionInsight Manager, choose **Cluster** > **Name of the desired cluster** > **Services** > **ZooKeeper**, and click the **Configurations** tab and then **All Configurations**. On the displayed page, search for the **clientPort** parameter and record its value.

Step 8 Create a topic.

sh kafka-topics.sh --create --topic *Topic name* **--partitions** *Number of partitions occupied by the topic* **--replication-factor** *Number of replicas of the topic* **--zookeeper** *IP address of the node where the ZooKeeper instance resides:clientPort***/kafka**

Step 9 Run the following command to view the topic information in the cluster:

sh kafka-topics.sh --list --zookeeper *IP address of the node where the ZooKeeper instance resides:clientPort***/kafka**

Step 10 Delete the topic created in **Step 8**.

sh kafka-topics.sh --delete --topic *Topic name* **--zookeeper** *IP address of the node where the ZooKeeper instance resides*:*clientPort***/kafka**

----End

12.14.2 Managing Kafka Topics

Scenario

You can manage Kafka topics on a cluster client based on service requirements. Management permission is required for clusters with Kerberos authentication enabled.

Prerequisites

You have installed the Kafka client.

Procedure

Step 1 Access the ZooKeeper instance page.

- For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **ZooKeeper** > **Instances**.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Instance**.

Step 2 View the IP addresses of the ZooKeeper role instance.

Record any IP address of the ZooKeeper instance.

Step 3 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see **Using an MRS Client**.

Step 4 Run the following command to switch to the client directory, for example, /**opt/client/Kafka/kafka/bin**.

cd /opt/client/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 6 Run the following command to perform user authentication (skip this step in normal mode):

kinit *Component service user*

Step 7 For versions earlier than MRS 3.x, run the following commands to manage Kafka topics:

- Creating a topic

  sh kafka-topics.sh --create --topic *Topic name* **--partitions** *Number of partitions occupied by the topic* **--replication-factor** *Number of replicas of the topic* **--zookeeper** *IP address of the node where the ZooKeeper instance resides:clientPort*/**kafka**

- Deleting a topic

  sh kafka-topics.sh --delete --topic *Topic name* **--zookeeper** *IP address of the node where the ZooKeeper instance resides*:*clientPort*/**kafka**

NOTE

- The number of topic partitions or topic backup replicas cannot exceed the number of Kafka instances.
- By default, the value of **clientPort** of ZooKeeper is **2181**.
- There are three ZooKeeper instances. Use the IP address of any one.
- For details about managing messages in Kafka topics, see **Managing Messages in Kafka Topics**.

Step 8 MRS 3.*x* and later versions: Use **kafka-topics.sh** to manage Kafka topics.

- Creating a topic:

  By default, partitions of a topic are distributed based on the number of partitions on the node and disk. To distribute partitions based on the disk capacity, set **log.partition.strategy** to **capacity** for the Kafka service.

  When a topic is created in Kafka, partitions and copies can be generated based on the combination of rack awareness and cross-AZ feature. The **--zookeeper** and **--bootstrap-server** modes are supported.

  – Disable the rack policy and cross-AZ feature (default policy).

    Copies of topics created based on this policy are randomly allocated to any node in the cluster.

    ./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic***--zookeeper** *IP address of any ZooKeeper node:clientPort***/kafka**

    ./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic* **--bootstrap-server** *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties**

    If you use **--bootstrap-server** to create a topic, set **rack.aware.enable** and **az.aware.enable** to **false**.

  – Enable the rack policy and disable the cross-AZ feature.

    The leader of each partition of the topic created based on this policy is randomly allocated on the cluster node. However, different replicas of the same partition are allocated to different racks. Therefore, when this policy is used, ensure that the number of nodes in each rack is the same, otherwise, the load of nodes in the rack with fewer nodes is much higher than the average load of the cluster.

    ./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic* **--zookeeper** *IP address of any ZooKeeper node:clientPort***/kafka --enable-rack-aware**

    ./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic* **--bootstrap-server** *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties**

    If you use **--bootstrap-server** to create a topic, set **rack.aware.enable** to **true** and **az.aware.enable** to **false**.

  – Disable the rack policy and enable the cross-AZ feature.

    The leader of each partition of the topic created based on this policy is randomly allocated on the cluster node. However, different replicas of the

same partition are allocated to different AZs. Therefore, when this policy is used, ensure that the number of nodes in each AZ is the same, otherwise, the load of nodes in the AZ with fewer nodes is much higher than the average load of the cluster.

./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic* **--zookeeper** *IP address of any ZooKeeper node:clientPort***/kafka --enable-az-aware**

./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic* **--bootstrap-server** *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties**

If you use **--bootstrap-server** to create a topic, set **rack.aware.enable** to **false** and **az.aware.enable** to **true**.

- Enable the rack policy and cross-AZ feature.

  The leader of each partition of the topic created based on this policy is randomly allocated on the cluster node. However, different replicas of the same partition are allocated to different racks in different AZs. This policy ensures that the number of nodes on each rack in each AZ is the same, otherwise, the load in the cluster is unbalanced.

  ./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic* **--replication-factor** *number of replicas of the topic* **--zookeeper** *IP address of any ZooKeeper node:clientPort***/kafka --enable-rack-aware --enable-az-aware**

  ./kafka-topics.sh --create --topic *topic name* **--partitions** *number of partitions occupied by the topic***--replication-factor** *number of replicas of the topic* **--bootstrap-server** *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties**

  If you use **--bootstrap-server** to create a topic, set **rack.aware.enable** and **az.aware.enable** to **true**.

  NOTE

  - Kafka supports topic creation in either of the following modes:
    - In **--zookeeper** mode, the client generates a copy allocation scheme. The community supports this mode from the beginning. To reduce the dependency on the ZooKeeper component, the community will delete the support for this mode in later versions. When creating a topic in this mode, you can select a copy allocation policy by combining the **--enable-rack-aware** and **--enable-az-aware** options. Note: The **--enable-az-aware** option can be used only when the cross-AZ feature is enabled on the server, that is, **az.aware.enable** is set to **true**. Otherwise, the execution fails.
    - In **--bootstrap-server** mode, the server generates a copy allocation solution. In later versions, the community supports only this mode for topic management. When a topic is created in this mode, the **--enable-rack-aware** and **--enable-az-aware** options cannot be used to control the copy allocation policy. The **rack.aware.enable** and **az.aware.enable** parameters can be used together to control the copy allocation policy. Note that the **az.aware.enable** parameter cannot be modified; if the cross-AZ feature is enabled during cluster creation, this parameter is automatically set to **true**; the **rack.aware.enable** parameter can be customized.

- List of topics:
  - **./kafka-topics.sh --list --zookeeper** *service IP address of any ZooKeeper node:clientPort*/**kafka**
  - **./kafka-topics.sh --list --bootstrap-server** *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties**
- Viewing the topic:
  - **./kafka-topics.sh --describe --zookeeper** *service IP address of any ZooKeeper node:clientPort*/**kafka** --**topic** *topic name*
  - **./kafka-topics.sh --describe --bootstrap-server** *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties --topic** *topic name*
- Modifying a topic:
  - **./kafka-topics.sh --alter --topic** *topic name*--**config** *configuration item=configuration value* **--zookeeper** *service IP address of any ZooKeeper node:clientPort*/**kafka**
- Expanding partitions:
  - **./kafka-topics.sh --alter --topic** *topic name* **--zookeeper** *service IP address of any ZooKeeper node:clientPort*/**kafka --command-config Kafka/kafka/config/client.properties --partitions** *number of partitions after the expansion*
  - **./kafka-topics.sh --alter --topic** *topic name* **--bootstrap-server** *IP address of the Kafka cluster:21007* **--command-config Kafka/kafka/config/client.properties --partitions** *number of partitions after the expansion*
- Deleting a topic
  - **./kafka-topics.sh --delete --topic** *topic name* **--zookeeper** *Service IP address of any ZooKeeper node:clientPort*/**kafka**
  - **./kafka-topics.sh --delete --topic** *topic name*--**bootstrap-server** *IP address of the Kafka cluster:21007* **--command-config ../config/client.properties**

----End

12.14.3 Querying Kafka Topics

Scenario

You can query existing Kafka topics on MRS.

Procedure

Step 1 Go to the Kafka service page.

- For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **Kafka**.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka**.

Step 2 Click **KafkaTopicMonitor**.

All topics are displayed in the list by default. You can view the number of partitions and replicas of the topics.

Step 3 Click the desired topic in the list to view its details.

----End

12.14.4 Managing Kafka User Permissions

Scenario

For clusters with Kerberos authentication enabled, using Kafka requires relevant permissions. MRS clusters can grant the use permission of Kafka to different users.

Table 12-258 lists the default Kafka user groups.

NOTE

In MRS 3.*x* or later, Kafka supports two types of authentication plug-ins: Kafka open source authentication plug-in and Ranger authentication plug-in.

This section describes the user permission management based on the Kafka open source authentication plug-in. For details about how to use the Ranger authentication plug-in, see **Adding a Ranger Access Permission Policy for Kafka**.

Table 12-258 Default Kafka user groups

User Group	Description
kafkaadmin	Kafka administrator group. Users in this group have the permissions to create, delete, read, and write all topics, and authorize other users.
kafkasuperuser	Kafka super user group. Users in this group have the permissions to read and write all topics.
kafka	Kafka common user group. Users in this group can access a topic only when they are granted with the read and write permissions of the topic by a user in the **kafkaadmin** group.

Prerequisites

- You have installed the Kafka client.
- A user in the **kafkaadmin** group, for example **admin**, has been prepared.

Procedure

Step 1 Access the ZooKeeper instance page.

- For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **ZooKeeper** > **Instances**.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services** > **ZooKeeper** > **Instance**.

Step 2 View the IP addresses of the ZooKeeper role instance.

Record the IP address of any ZooKeeper instance.

Step 3 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see **Using an MRS Client**.

Step 4 Run the following command to switch to the client directory, for example, **/opt/client/Kafka/kafka/bin**.

cd /opt/client/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 6 Run the following command to authenticate the user(skip this step in normal mode):

kinit *Component service user*

Step 7 Versions earlier than MRS 3.x: Select the scenario required by the service and manage Kafka user permissions.

- Querying the permission list of a topic

  sh kafka-acls.sh --authorizer-properties zookeeper.connect=*IP address of the node where the ZooKeeper instance resides*:2181/kafka --list --topic *Topic name*

- Adding producer permission to a user

  sh kafka-acls.sh --authorizer-properties zookeeper.connect=*IP address of the node where the ZooKeeper instance resides*:2181/kafka --add --allow-principal User:*Username* --producer --topic *Topic name*

- Removing producer permission of a user

  sh kafka-acls.sh --authorizer-properties zookeeper.connect=*IP address of the node where the ZooKeeper instance resides*:2181/kafka --remove --allow-principal User:*Username* --producer --topic *Topic name*

- Adding consumer permission to a user

  sh kafka-acls.sh --authorizer-properties zookeeper.connect=*IP address of the node where the ZooKeeper instance resides*:2181/kafka --add --allow-principal User:*Username* --consumer --topic *Topic name* --group *Consumer group name*

- Removing consumer permission of a user

  sh kafka-acls.sh --authorizer-properties zookeeper.connect=*IP address of the node where the ZooKeeper instance resides*:2181/kafka --remove --allow-principal User:*Username* --consumer --topic *Topic name* --group *Consumer group name*

  NOTE

  You need to enter **y** twice to confirm the removal of permission.

Step 8 MRS 3.*x* and later versions: The following table lists the common commands used for user authorization when **kafka-acl.sh** is used.

- View the permission control list of a topic:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--list --topic** *<Topic name>*

  ./kafka-acls.sh --bootstrap-server <*IP address of the Kafkacluster:21007*> **--command-config ../config/client.properties --list --topic** <*topic name*>

- Add the Producer permission for a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--add --allow-principal User:***<User name>* **--producer --topic** *<Topic name>*

  ./kafka-acls.sh --bootstrap-server <*IP address of the Kafkacluster:21007*> **--command-config ../config/client.properties --add --allow-principal User:***<username>* **--producer --topic** <*topic name*>

- Assign the Producer permission to a user in batches.

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--add --allow-principal User:***<User name>* **--producer --topic** *<Topic name>* **--resource-pattern-type prefixed**

  ./kafka-acls.sh --bootstrap-server <*IP address of the Kafkacluster:21007*> **--command-config ../config/client.properties --add --allow-principal User:***<username>* **--producer --topic** *<topic name>***--resource-pattern-type prefixed**

- Remove the Producer permission from a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--remove --allow-principal User:***<User name>* **--producer --topic** *<Topic name>*

  ./kafka-acls.sh --bootstrap-server <*IP address of the Kafkacluster:21007*> **--command-config ../config/client.properties --remove --allow-principal User:***<username>* **--producer --topic** <*topic name*>

- Delete the Producer permission of a user in batches:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--remove --allow-principal User:***<User name>* **--producer --topic** *<Topic name>* **--resource-pattern-type prefixed**

  ./kafka-acls.sh --bootstrap-server <*IP address of the Kafkacluster:21007*> **--command-config ../config/client.properties --remove --allow-principal User:***<username>* **--producer --topic** *<topic name>***--resource-pattern-type prefixed**

- Add the Consumer permission for a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--add --allow-principal User:***<User name>* **--consumer --topic** *<Topic name>* **--group** *<Consumer group name>*

  ./kafka-acls.sh --bootstrap-server *<IP address of the Kafkacluster:21007>* **--command-config ../config/client.properties --add --allow-principal User:***<username>* **--consumer --topic** *<topicname>* **--group** *<consumer group name>*

- Add consumer permissions to a user in batches:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper node:21812181/kafka >* **--add --allow-principal User:***<User name>* **--consumer --topic** *<Topic name>* **--group** *<Consumer group name>* **--resource-pattern-type prefixed**

  ./kafka-acls.sh --bootstrap-server *<IP address of the Kafkacluster:21007>* **--command-config ../config/client.properties --add --allow-principal User:***<username>* **--consumer --topic** *<topicname>* **--group** *<consumer group name>* **--resource-pattern-type prefixed**

- Remove the consumer permission from a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*<Service IP address of any ZooKeeper nodeIP:21812181/kafka >* **--remove --allow-principal User:***<User name>* **--consumer --topic** *<Topic name>* **--group** *<Consumer group name>*

  ./kafka-acls.sh --bootstrap-server *<IP address of the Kafkacluster:21007>* **--command-config ../config/client.properties --remove --allow-principal User:***<username>* ***--consumer --topic*** *<topic name>* **--group** *<consumer group name>*

- Delete the consumer permission of a user in batches:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=*Service IP address of any ZooKeeper node:21812181/kafka >* **--remove --allow-principal User:***<User name>* **--consumer --topic** *<Topic name>* **--group** *<Consumer group name>* **--resource-pattern-type prefixed**

  ./kafka-acls.sh --bootstrap-server *<IP address of the Kafkacluster:21007>* **--command-config ../config/client.properties --remove --allow-principal User:***<username>* **--consumer --topic** *<topicname>* **--group** *<consumer group name>* **--resource-pattern-type prefixed**

----End

12.14.5 Managing Messages in Kafka Topics

Scenario

You can produce or consume messages in Kafka topics using the MRS cluster client. For clusters with Kerberos authentication enabled, you must have the permission to perform these operations.

Prerequisites

You have installed the Kafka client.

Procedure

Step 1 Go to the Kafka service page.

- For versions earlier than MRS 3.x, click the cluster name to go to the cluster details page and choose **Components** > **Kafka**.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka**.

Step 2 Click **instance**. Query the IP addresses of the Kafka instances.

Record the IP address of any Kafka instance.

Step 3 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see **Using an MRS Client**.

Step 4 Run the following command to switch to the client directory, for example, /**opt/client/Kafka/kafka/bin**.

cd /opt/client/Kafka/kafka/bin

Step 5 Run the following command to configure environment variables:

source /opt/client/bigdata_env

Step 6 For clusters with Kerberos authentication enabled, run the following command to authenticate the user. For clusters with Kerberos authentication disabled, skip this step.

kinit *Kafka user*

Example:

kinit admin

Step 7 Manage messages in Kafka topics using the following commands:

- Producing messages

  sh kafka-console-producer.sh --broker-list *IP address of the node where the Kafka instance resides*:**9092 --topic** *Topic name* **--producer.config /opt/client/Kafka/kafka/config/producer.properties**

  You can input specified information as the messages produced by the producer and then press **Enter** to send the messages. To end message producing, press **Ctrl + C** to exit.

- Consuming messages

  sh kafka-console-consumer.sh --topic *Topic name* **--bootstrap-server** *IP address of the node where the Kafka instance resides*:**9092 --consumer.config /opt/client/Kafka/kafka/config/consumer.properties**

In the configuration file, **group.id** (indicating the consumer group) is set to **example-group1** by default. Users can change the value as required. The value takes effect each time consumption occurs.

By default, the system reads unprocessed messages in the current consumer group when the command is executed. If a new consumer group is specified in the configuration file and the **--from-beginning** parameter is added to the command, the system reads all messages that have not been automatically deleted in Kafka.

NOTE

----End

12.14.6 Synchronizing Binlog-based MySQL Data to the MRS Cluster

This section describes how to use the Maxwell data synchronization tool to migrate offline binlog-based data to an MRS Kafka cluster.

Maxwell is an open source application that reads MySQL binlogs, converts operations, such as addition, deletion, and modification, into a JSON format, and sends them to an output end, such as a console, a file, and Kafka. For details about Maxwell, visit **https://maxwells-daemon.io**. Maxwell can be deployed on a MySQL server or on other servers that can communicate with MySQL.

Maxwell runs on a Linux server, including EulerOS, Ubuntu, Debian, CentOS, and OpenSUSE. Java 1.8+ must be supported.

The following provides details about data synchronization.

1. **Configuring MySQL**
2. **Installing Maxwell**
3. **Configuring Maxwell**
4. **Starting Maxwell**
5. **Verifying Maxwell**
6. **Stopping Maxwell**
7. **Format of the Maxwell Generated Data and Description of Common Fields**

Configuring MySQL

Step 1 Start the binlog, open the **my.cnf** file in MySQL, and check whether **server_id**, **log-bin**, and **binlog_format** are configured in the **[mysqld]** block. If they are not configured, run the following command to add configuration items and restart MySQL. If they are configured, skip this step.

```
$ vi my.cnf

[mysqld]
server_id=1
log-bin=master
binlog_format=row
```

Step 2 Maxwell needs to connect to MySQL, create a database named **maxwell** for storing metadata, and access the database to be synchronized. Therefore, you are

advised to create a MySQL user for Maxwell to use. Log in to MySQL as user **root** and run the following commands to create a user named **maxwell** (*XXXXXX* indicates the password and needs to be replaced with actual one).

- If Maxwell is deployed on a non-MySQL server, the created user **maxwell** must have a permission to remotely log in to the database. In this case, run the following command to create the user:

  mysql> GRANT ALL on maxwell.* to 'maxwell'@'%' identified by 'XXXXXX';

  mysql> GRANT SELECT, REPLICATION CLIENT, REPLICATION SLAVE on *.* to 'maxwell'@'%';

- If Maxwell is deployed on the MySQL server, the created user **maxwell** can be configured to log in to the database only on the local host. In this case, run the following command:

  mysql> GRANT SELECT, REPLICATION CLIENT, REPLICATION SLAVE on *.* to 'maxwell'@'localhost' identified by 'XXXXXX';

  mysql> GRANT ALL on maxwell.* to 'maxwell'@'localhost';

----End

Installing Maxwell

Step 1 Download the installation package at **https://github.com/zendesk/maxwell/releases** and select the **maxwell-XXX.tar.gz** binary file for download. In the file name, **XXX** indicates a version number.

Step 2 Upload the **tar.gz** package to any directory (the **/opt** directory of the Master node used as an example here).

Step 3 Log in to the server where Maxwell is deployed and run the following command to go to the directory where the **tar.gz** package is stored.

cd /opt

Step 4 Run the following commands to decompress the **maxwell-XXX.tar.gz** package and go to the **maxwell-XXX** directory:

tar -zxvf maxwell-XXX.tar.gz

cd maxwell-XXX

----End

Configuring Maxwell

If the **conf** directory exists in the **maxwell-XXX** folder, configure the **config.properties** file. For details about the configuration items, see **Table 12-259**. If the **conf** directory does not exist, change **config.properties.example** in the **maxwell-XXX** folder to **config.properties**.

Table 12-259 Maxwell configuration item description

Parameter	Mandatory	Description	Default Value
user	Yes	Name of the user for connecting to MySQL, that is, the user created in **Step 2**.	-
password	Yes	Password for connecting to MySQL	-
host	No	MySQL address	localhost
port	No	MySQL port	3306
log_level	No	Log print level. The options are as follows: • debug • info • warn • error	info
output_ddl	No	Whether to send a DDL (modified based on definitions of the database and data table) event • **true**: Send DDL events. • **false**: Do not send DDL events.	false
producer	Yes	Producer type. Set this parameter to **kafka**. • **stdout**: Log the generated events. • **kafka**: Send the generated events to Kafka.	stdout
producer_partition_by	No	Partition policy used to ensure that data of the same type is written to the same partition of Kafka. • **database**: Events of the same database are written to the same partition of Kafka. • **table**: Events of the same table are written to the same partition of Kafka.	databa
ignore_producer_error	No	Specifies whether to ignore the error that the producer fails to send data. • **true**: The error information is logged and the error data is skipped. The program continues to run. • **false**: The error information is logged and the program is terminated.	true
metrics_slf4j_interval	No	Interval for outputting statistics on data successfully uploaded or failed to be uploaded to Kafka in logs. The unit is second.	60

Parameter	Mandatory	Description	Default Value
kafka.bootstrap.servers	Yes	Address of the Kafka proxy node. The value is in the format of **HOST:PORT[,HOST:PORT]**.	-
kafka_topic	No	Name of the topic that is written to Kafka	maxwell
dead_letter_topic	No	Kafka topic used to record the primary key of the error log record when an error occurs when the record is sent	-
kafka_version	No	Kafka producer version used by Maxwell, which cannot be configured in the **config.properties** file. You need to use the **--kafka_version xxx** parameter to import the version number when starting the command.	-
kafka_partition_hash	No	Kafka topic partitioning algorithm. The value can be **default** or **murmur3**.	default
kafka_key_format	No	Key generation method of the Kafka record. The value can be **array** or **Hash**.	Hash
ddl_kafka_topic	No	Topic that is written to the DDL operation when **output_ddl** is set to **true**	{kafka_topic}
filter	No	Used to filter databases or tables. ● If only the **mydatabase** database needs to be collected, set this parameter to the following: exclude: *.*,include: mydatabase.* ● If only the **mydatabase.mytable** table needs to be collected, set this parameter to the following: exclude: *.*,include: mydatabase.mytable ● If only the **mytable**, **mydate_123**, and **mydate_456** tables in the **mydatabase** database need to be collected, set this parameter to the following: exclude: *.*,include: mydatabase.mytable,include: mydatabase./mydate_\\d*/	-

Starting Maxwell

Step 1 Log in to the server where Maxwell is deployed.

Step 2 Run the following command to go to the Maxwell installation directory:

cd /opt/maxwell-1.21.0/

NOTE

For the first time to use Maxwell, you are advised to change **log_level** in **conf/config.properties** to **debug** (debug level) so that you can check whether data can be obtained from MySQL and sent to Kafka after startup. After the entire process is debugged, change **log_level** to **info**, and then restart Maxwell for the modification to take effect.

```
# log level [debug | info | warn | error]
log_level=debug
```

Step 3 Run the following commands to start Maxwell:

source /opt/client/bigdata_env

bin/Maxwell

**bin/maxwell --user='maxwell' --password='XXXXXX' --host='127.0.0.1' **

--producer=kafka --kafka.bootstrap.servers=kafkahost:9092 --kafka_topic=Maxwell

In the preceding commands, **user**, **password**, and **host** indicate the username, password, and IP address of MySQL, respectively. You can configure the three parameters by modifying configurations of the configuration items or using the preceding commands. **kafkahost** indicates the IP address of the Core node in the streaming cluster.

If information similar to the following appears, Maxwell has started successfully:

```
Success to start Maxwell [78092].
```

----End

Verifying Maxwell

Step 1 Log in to the server where Maxwell is deployed.

Step 2 View the logs. If the log file does not contain an ERROR log and the following information is displayed, the connection between Maxwell and MySQL is normal:

```
BinlogConnectorLifecycleListener - Binlog connected.
```

Step 3 Log in to the MySQL database and update, create, or delete test data. The following provides operation statement examples for your reference.

```
--Creating a database
create database test;
--Creating a table
create table test.e (
  id int(10) not null primary key auto_increment,
  m double,
  c timestamp(6),
  comment varchar(255) charset 'latin1'
);
-- Adding a record
insert into test.e set m = 4.2341, c = now(3), comment = 'I am a creature of light.';
--Updating a record
update test.e set m = 5.444, c = now(3) where id = 1;
--Deleting a record
delete from test.e where id = 1;
--Modifying a table
alter table test.e add column torvalds bigint unsigned after m;
--Deleting a table
drop table test.e;
-- Deleting a database
drop database test;
```

Step 4 Check the Maxwell logs. If no WARN/ERROR is displayed, Maxwell is installed and configured properly.

To check whether the data is successfully uploaded, set **log_level** in the **config.properties** file to **debug**. When the data is successfully uploaded, the following JSON data is printed immediately. For details about the fields, see **Format of the Maxwell Generated Data and Description of Common Fields**.

```
{"database":"test","table":"e","type":"insert","ts":1541150929,"xid":60556,"commit":true,"data":
{"id":1,"m":4.2341,"c":"2018-11-02 09:28:49.297000","comment":"I am a creature of light."}}
.....
```

NOTE

After the entire process is debugged, you can change the value of **log_level** in the **config.properties** file to **info** to reduce the number of logs to be printed and restart Maxwell for the modification to take effect.

```
# log level [debug | info | warn | error]
log_level=info
```

----End

Stopping Maxwell

Step 1 Log in to the server where Maxwell is deployed.

Step 2 Run the command to obtain the Maxwell process ID (PID). The second field in the command output is PID.

ps -ef | grep Maxwell | grep -v grep

Step 3 Run the following command to forcibly stop the Maxwell process:

kill -9 PID

----End

Format of the Maxwell Generated Data and Description of Common Fields

The data generated by Maxwell is in JSON format. The common fields are described as follows:

- **type**: operation type. The options are **database-create**, **database-drop**, **table-create**, **table-drop**, **table-alter**, **insert**, **update**, and **delete**.
- **database**: name of the database to be operated
- **ts**: operation time, which is a 13-digit timestamp
- **table**: name of the table to be operated
- **data**: content after data is added, deleted, or modified
- **old**: content before data is modified or schema definition before a table is modified
- **sql**: SQL statement for DDL operations
- **def**: schema definition for table creation and modification
- **xid**: unique ID of an object
- **commit**: check whether such operations as data addition, deletion, and modification have been submitted.

12.14.7 Creating a Kafka Role

Scenario

This section describes how to create and configure a Kafka role.

This section applies to MRS 3.x or later.

NOTE

Users can create Kafka roles only in security mode.

If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Kafka**.

Prerequisites

The system administrator has understood the service requirements.

Procedure

Step 1 Log in to FusionInsight Manager and choose **System** > **Permission** > **Role**.

Step 2 On the displayed page, click **Create Role** and enter a **Role Name** and **Description**.

Step 3 On the **Configure Resource Permission** page, choose *Name of the desired cluster* > **Kafka**.

Step 4 Select permissions based on service requirements. For details about configuration items, see **Table 12-260**.

Table 12-260 Description

Scenario	Role Authorization
Setting the Kafka administrator permissions	In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Kafka** > **Kafka Manager Privileges**. **NOTE** This permission allows you to create and delete topics, but does not allow you to produce or consume any topics.
Setting the production permission of a user on a topic	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Kafka** > **Kafka Topic Producer And Consumer Privileges**. 2. In the **Permission** column of the specified topic, select **Kafka Producer Permission**.

Scenario	Role Authorization
Setting the consumption permission of a user on a topic	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Kafka** > **Kafka Topic Producer And Consumer Privileges**. 2. In the **Permission** column of the specified topic, select **Kafka Consumer Privileges**.

Step 5 Click **OK**, and return to the **Role** page.

----End

12.14.8 Kafka Common Parameters

This section applies to MRS 3.x or later.

Navigation path for setting parameters:

For details about how to set parameters, see **Modifying Cluster Service Configuration Parameters**.

Common Parameters

Table 12-261 Parameter description

Parameter	Description	Default Value
log.dirs	List of Kafka data storage directories. Use commas (,) to separate multiple directories.	%{@auto.detect.datapart.bk.log.logs}
KAFKA_HEAP_OPTS	Specifies the JVM option used for Kafka to start broker. It is recommended that you set this parameter based on service requirements.	-Xmx6G -Xms6G
auto.create.topics.enable	Indicates whether a topic is automatically created. If this parameter is set to **false**, you need to run a command to create a topic before sending a message.	true
default.replication.factor	Default number of replicas of a topic is automatically created.	2

Parameter	Description	Default Value
monitor.preInitDelay	Delay of the first health check after the server is started. If the startup takes a long time, increase the value of the parameter. Unit: millisecond	600,000

Timeout Parameters

Table 12-262 Broker-related timeout parameters

Parameter	Description	Default Value	Impact
controller.socket.timeout.ms	Specifies the timeout for connecting controller to broker. Unit: millisecond	30,000	Generally, retain the default value of this parameter.
group.max.session.timeout.ms	Specifies the maximum session timeout during the consumer registration. Unit: millisecond	1800000	The configured value must be less than the value of this parameter.
group.min.session.timeout.ms	Specifies the minimum session timeout during the consumer registration. Unit: millisecond	6,000	The configured value must be greater than the value of this parameter.
offsets.commit.timeout.ms	Specifies the timeout for the Offset to submit requests. Unit: millisecond	5,000	This parameter specifies the maximum delay for processing an Offset request.
replica.socket.timeout.ms	Specifies the timeout of the request for synchronizing replica data. Its value must be greater than or equal to that of the **replica.fetch.wait.max.ms** parameter. Unit: millisecond	30,000	Specifies the maximum timeout for establishing a channel before the synchronization thread sends a synchronization request. The value must be greater than that of the **replica.fetch.wait.max.ms** parameter.

Parameter	Description	Default Value	Impact
request.timeout.ms	Specifies the timeout for waiting for a response after the client sends a connection request. If no response is received within the timeout, the client resends the request. A request failure is returned after the maximum retry times is reached. Unit: millisecond	30,000	This parameter is configured when the networkclient connection is transferred in the controller and replica threads on the broker node.
transaction.max.timeout.ms	Specifies the maximum timeout allowed by the transaction. If the client request time exceeds the value of this parameter, broker returns an error in InitProducerIdRequest. This prevents a long client request timeout, ensuring that consumer can receive topics. Unit: millisecond	900,000	Specifies the maximum timeout for transactions.
user.group.cache.timeout.sec	Specifies the time when the user group information is stored in the cache. Unit: second	300	Specifies the time for caching the mapping between users and user groups. If time exceeds the threshold, the system automatically runs the **id -Gn** command to query the user information. During this period, the mapping in the cache is used.
zookeeper.connection.timeout.ms	Specifies the timeout for connecting to ZooKeeper. Unit: millisecond	45,000	This parameter specifies the duration for connecting the ZooKeeper and zkclient for the first time. If the duration exceeds the value of this parameter, the zkclient automatically disconnects the connection.

Parameter	Description	Default Value	Impact
zookeeper.session.timeout.ms	Specifies the ZooKeeper session timeout duration. During this period, ZooKeeper disconnects the connection if broker does not report its heartbeats to ZooKeeper. Unit: millisecond	45,000	ZooKeeper session timeout has the following functions: 1) Based on value of this parameter and the number of ZooKeeper URLs in ZKURL, if the connection duration exceeds the node timeout value (sessionTimeout/ Number of transferred ZooKeeper URLs), the connection fails and the system attempts to connect to the next node. 2) After the connection is established, a session (for example, the temporary BrokerId node registered on the ZooKeeper) is cleared by the ZooKeeper a session timeout later if the broker is stopped.

Table 12-263 Producer-related timeout parameters

Parameter	Description	Default Value	Impact
request.timeout.ms	Specifies the timeout of a message request.	30,000	If a network fault occurs, increase the value of this parameter. If the value is too small, the Batch Expire occurs.

Table 12-264 Consumer-related timeout parameters

Parameter	Description	Default Value	Impact
connections.max.idle.ms	Specifies the maximum retention period for idle connections.	600,000	If the idle connection time is greater than this parameter value, this connection is disconnected. If necessary, a new connection is created.
request.timeout.ms	Specifies the timeout for consumer requests.	30,000	If the request times out, the request will fail and be sent again.

12.14.9 Safety Instructions on Using Kafka

This section applies to MRS 3.*x* or later.

Brief Introduction to Kafka APIs

- Producer API

  Indicates the API defined in **org.apache.kafka.clients.producer.KafkaProducer**. When **kafka-console-producer.sh** is used, the API is used by default.

- Consumer API

  Indicates the API defined in **org.apache.kafka.clients.consumer.KafkaConsumer**. When **kafka-console-consumer.sh** is used, the API is used by default.

NOTE

In MRS 3.*x* or later, Kafka no longer support old Producer or Consumer APIs.

Protocol Description for Accessing Kafka

The protocols used to access Kafka are as follows: PLAINTEXT, SSL, SASL_PLAINTEXT, and SASL_SSL.

When Kafka service is started, the listeners using the PLAINTEXT and SASL_PLAINTEXT protocols are started. You can set **ssl.mode.enable** to **true** in Kafka service configuration to start listeners using SSL and SASL_SSL protocols. The following table describes the four protocols:

Protocol	Description	Default Port
PLAINTEXT	Supports plaintext access without authentication.	9092

Protocol	Description	Default Port
SASL_PLAINTEXT	Supports plaintext access with Kerberos authentication.	21007
SSL	Supports SSL-encrypted access without authentication.	9093
SASL_SSL	Supports SSL-encrypted access with Kerberos authentication.	21009

ACL Settings for a Topic

To view and set topic permission information, run the **kafka-acls.sh** script on the Linux client. For details, see **Managing Kafka User Permissions**.

Use of Kafka APIs in Different Scenarios

- Scenario 1: accessing the topic with an ACL

Used API	User Group	Client Parameter	Server Parameter	Accessed Port
API	Users need to meet one of the following conditions: • In the administrator group • In the **kafkaadmin** group • In the **kafkasuperuser** group • In the **kafka** group and be authorized	security.inter.broker.protocol=SASL_PLAINTEXT sasl.kerberos.service.name = kafka	-	sasl.port (The default number is 21007.)
		security.protocol=SASL_SSL sasl.kerberos.service.name = kafka	Set **ssl.mode.enable** to **true**.	sasl-ssl.port (The default number is 21009.)

- Scenario 2: accessing the topic without an ACL

Used API	User Group	Client Parameter	Server Parameter	Accessed Port
API	Users need to meet one of the following conditions: • In the administrator group • In the **kafkaadmin** group • In the **kafkasuperuser** group	security.protocol=SASL_PLAINTEXT sasl.kerberos.service.name = kafka	-	sasl.port (The default number is 21007.)
	Users are in the **kafka** group.		Set **allow.everyone.if.no.acl.found** to **true**. **NOTE** In normal mode, the server parameter **allow.everyone.if.no.acl.found** does not need to be modified.	sasl.port (The default number is 21007.)
	Users need to meet one of the following conditions: • In the administrator group • In the **kafkaadmin** group • In the **kafkasuperuser** group	security.protocol=SASL_SSL sasl.kerberos.service.name = kafka	Set **ssl.mode.enable** to **true**.	sasl-ssl.port (The default number is 21009.)

Used API	User Group	Client Parameter	Server Parameter	Accessed Port
	Users are in the **kafka** group.		1. Set **allow.everyone.if.no.acl.found** to **true**. 2. Set **ssl.mode.enable** to **true**.	sasl-ssl.port (The default number is 21009.)
	-	security.protocol=PLAINTEXT	Set **allow.everyone.if.no.acl.found** to **true**.	port (The default number is 9092.)
	-	security.protocol=SSL	1. Set **allow.everyone.if.no.acl.found** to **true**. 2. Set **ssl.mode.enable** to **true**.	ssl.port (The default number is 9063.)

12.14.10 Kafka Specifications

This section applies to MRS 3.x or later.

Upper Limit of Topics

The maximum number of topics depends on the number of file handles (mainly used by data and index files on site) opened in the process.

1. Run the **ulimit -n** command to view the maximum number of file handles that can be opened in the process.
2. Run the **lsof -p** *<Kafka PID>* command to view the file handles (which may keep increasing) that are opened in the Kafka process on the current single node.
3. Determine whether the maximum number of file handles will be reached and whether the running of Kafka is affected after required topics are created, and estimate the maximum size of data that each partition folder can store and the number of data (*.log file, whose default size is 1 GB and can be adjusted by modifying **log.segment.bytes**) and index (*.index file, whose default size is 10 MB and can be adjusted by modifying **log.index.size.max.bytes**) files that will be produced after required topics are created.

Number of Concurrent Consumers

In an application, it is recommended that the number of concurrent consumers in a group be the same as the number of partitions in a topic, ensuring that a consumer consumes data in only a specified partition. If the number of concurrent consumers is more than the number of partitions, the redundant consumers have no data to consume.

Relationship Between Topic and Partition

- If *K* Kafka nodes are deployed in the cluster, each node is configured with *N* disks, the size of each disk is *M*, the cluster contains *n* topics (named as T1, T2, ..., Tn), the data input traffic per second of the *m* topic is *X* (Tm) MB/s, the number of configured replicas is *R* (Tm), and the configured data retention time is *Y* (Tm) hour, the following requirement must be met:

$$M \times N \times K > \sum_{i=T1}^{Tn} (X(i)R(i)Y(i) \times 3600)$$

- If the size of a disk is *M*, the disk has *n* partitions (named as P0, P1, ..., Pn), the data write traffic per second of the *m* partition is *Q* (Pm) MB/s (calculation method: data traffic of the topic to which the *m* partition belongs divided by the number of partitions), and the data retention time is *T* (Pm) hours, the following requirement must be met for the disk:

$$M > \sum_{i=P0}^{Pn} (Q(i)T(i) \times 3600)$$

- Based on the throughput, if the throughput that can be reached by the producer is *P*, the throughput that can be reached by the consumer is *C*, and the expected throughput of Kafka is *T*, it is recommended that the number of partitions of the topic be set to Max(T/P, T/C).

NOTE

- In a Kafka cluster, more partitions mean higher throughput. However, too many partitions also pose potential impacts, such as a file handle increase, unavailability increase (for example, if a node is faulty, the time window becomes large after the leader is reselected in some partitions), and end-to-end latency increase.
- Suggestion: The disk usage of a partition is smaller than or equal to 100 GB; the number of partitions on a node is smaller than or equal to 3,000; the number of partitions in the entire cluster is smaller than or equal to 10,000.

12.14.11 Using the Kafka Client

Scenario

This section guides users to use a Kafka client in an O&M or service scenario.

This section applies to MRS 3.x or later.

Prerequisites

- The client has been installed. For example, the installation directory is **/opt/client**.

- Service component users are created by the administrator as required. Machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login. (Not involved in normal mode)
- After changing the domain name of a cluster, redownload the client to ensure that the **kerberos.domain.name** value in the configuration file of the client is set to the correct server domain name.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to perform user authentication (skip this step in normal mode):

kinit *Component service user*

Step 5 Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

Step 6 Run the following command to use the client tool to view and use the help information:

- **./kafka-console-consumer.sh**: Kafka message reading tool
- **./kafka-console-producer.sh**: Kafka message publishing tool
- **./kafka-topics.sh**: Kafka topic management tool

----End

12.14.12 Configuring Kafka HA and High Reliability Parameters

Scenario

For the Kafka message transmission assurance mechanism, different parameters are available for meeting different performance and reliability requirements. This section describes how to configure Kafka high availability (HA) and high reliability parameters.

This section applies to MRS 3.x or later.

Impact on the System

- Impact of HA and high performance configurations:

NOTICE

After HA and high performance are configured, the data reliability decreases. Specifically, data may be lost of disks or nodes are faulty.

- Impact of high reliability configurations:
  - Deteriorated performance

    If **ack** is set to **–1**, data written is considered as successful only when data is written to multiple replicas. As a result, the delay of a single message increases and the client processing capability decreases. The impact is subject to the actual test data.
  - Reduced availability

    A replica that is not in the ISR list cannot be elected as a leader. If the leader goes offline and other replicas are not in the ISR list, the partition remains unavailable until the leader node recovers. When the node where a replica of a partition is located is faulty, the minimum number of successful replicas cannot be met. As a result, service writing fails.
- If parameters are at the service level, Kafka needs to be restarted. You are advised to modify the service-level configuration in the change window.

Parameter Description

- If services require high availability and high performance,

  set the parameters listed in **Table 12-265** on the server. For details about the parameter configuration entry, see **Modifying Cluster Service Configuration Parameters**.

  Table 12-265 Server HA and high performance parameters

Parameter	Default Value	Description
unclean.leader.election.enable	true	Specifies whether a replica that is not in the ISR can be selected as the leader. If this parameter is set to **true**, data may be lost.
auto.leader.rebalance.enable	true	Specifies whether the leader automated balancing function is used. If this parameter is set to **true**, the controller periodically balances the leader of each partition on all nodes and assigns the leader to a replica with a higher priority.
min.insync.replicas	1	Specifies the minimum number of replicas to which data is written when **acks** is set to **-1** for the Producer.

Set the parameters listed in **Table 12-266** in the client configuration file **producer.properties**. The path for storing **producer.properties** is /**opt/client/Kafka/kafka/config/producer.properties**, where **/opt/client** indicates the installation directory of the Kafka client.

Table 12-266 Client HA and high performance parameters

Parameter	Default Value	Description
acks	1	The leader needs to check whether the message has been received and determine whether the required operation has been processed. This parameter affects message reliability and performance. ● If this parameter is set to **0**, the producer does not wait for any response from the server, and the message is considered successful. ● If this parameter is set to **1**, when the leader of the replica verifies that data has been written into the cluster, the leader returns a response without waiting for data to be written to all replicas. In this case, if the leader is abnormal when the leader makes the confirmation but replica synchronization is not complete, data will be lost. ● If this parameter is set to **-1**, the message is considered to be successfully received only when all synchronized replicas are confirmed. If the **min.insync.replicas**

Parameter	Default Value	Description
		parameter is also configured, data can be written into multiple replicas. In this case, records will not be lost as long as one replica remains active.

- To ensure high data reliability for services,

  set the parameters listed in **Table 12-267** on the server. For details about the parameter configuration entry, see **Modifying Cluster Service Configuration Parameters**.

  Table 12-267 Server HA parameters

Parameter	Recommended Value	Description
unclean.leader.election.enable	false	A replica that is not in the ISR list cannot be elected as a leader.
min.insync.replicas	2	Specifies the minimum number of replicas to which data is written when **acks** is set to **-1** for the Producer. Ensure that the value of **min.insync.replicas** is equal to or less than that of **replication.factor**.

Set the parameters listed in **Table 12-268** in the client configuration file **producer.properties**. The path for storing **producer.properties** is **/opt/client/Kafka/kafka/config/producer.properties**, where **/opt/client** indicates the installation directory of the Kafka client.

Table 12-268 Server HA parameters

Parameter	Recommended Value	Description
acks	-1	The leader needs to check whether the message has been received and determine whether the required operation has been processed. If this parameter is set to **-1**, the message is considered to be successfully received only when all replicas in the ISR list have confirmed to receive the message. This parameter is used along with **min.insync.replicas** to ensure that multiple copies are successfully written. As long as one copy is active, the record will not be lost. If this parameter is set to **-1**, the production performance deteriorates. Therefore, you need to set this parameter based on the actual situation.

Configuration Suggestions

Configure parameters based on requirements on reliability and performance in the following service scenarios:

- For valued data, you are advised to configure RAID1 or RAID5 for Kafka data directory disks to improve data reliability when a single disk is faulty.
- For parameters that can be modified at the topic level, the service level configurations are used by default.

  These parameters can be separately configured based on topic reliability requirements. For example, log in to the Kafka client as user **root**, and run the following command to configure the reliability parameter with topic named test in the client installation directory:

  cd Kafka/kafka/bin

  kafka-topics.sh --zookeeper 192.168.1.205:2181/kafka --alter --topic test --config unclean.leader.election.enable=false --config min.insync.replicas=2

192.168.1.205 indicates the ZooKeeper service IP address.

- If parameters are at the service level, Kafka needs to be restarted. You are advised to modify the service-level configuration in the change window.

12.14.13 Changing the Broker Storage Directory

Scenario

This section applies to MRS 3.x or later.

When a broker storage directory is added, the system administrator needs to change the broker storage directory on FusionInsight Manager, to ensure that the Kafka can work properly. The new topic partition will be generated in the directory that has fewest partitions. Changing the ZooKeeper storage directory includes the following scenarios:

NOTE

Because Kafka does not detect disk capacity, ensure that the disk quantity and capacity configured for each Broker instance are the same.

- Change the storage directory of the Broker role. In this way, the storage directories of all Broker instances are changed.
- Change the storage directory of a single Broker instance. In this way, only the storage directory of this Broker instance is changed, and the storage directories of other Broker instances remain the same.

Impact on the System

- Changing the Broker role storage directory requires the restart of services. The services cannot be accessed during the restart.
- The storage directory of a single Broker instance can be changed only after the instance is restarted. The instance cannot provide services during the restart.
- The directory for storing service parameter configurations must also be updated.

Prerequisites

- New disks have been prepared and installed on each data node, and the disks are formatted.
- The Kafka client has been installed.
- When you change the storage directory of a single Broker instance, the number of active Broker instances must be greater than the number of backups specified during topic creation.

Procedure

Changing the storage directory of the Kafka role

Step 1 Log in as user **root** to each node on which the Kafka service is installed, and perform the following operations:

1. Create a target directory.

   For example, to create the target directory **${BIGDATA_DATA_HOME}/kafka/data2**, run the following command:

   mkdir ${BIGDATA_DATA_HOME}/kafka/data2

2. Mount the directory to the new disk. For example, mount **${BIGDATA_DATA_HOME}/kafka/data2** to the new disk.
3. Modify permissions on the new directory.

   For example, to modify permissions on the **${BIGDATA_DATA_HOME}/kafka/data2** directory, run the following commands:

   chmod 700 ${BIGDATA_DATA_HOME}/kafka/data2 -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/kafka/data2 -R**

Step 2 Log in to FusionInsight Manager for clusters of MRS 3.*x* or later and choose **Cluster** > **Services** > **Kafka** > **Configurations**.

Step 3 Add a new directory to the end of the default value of **log.dirs**.

Enter **log.dirs** in the search box and add the new directory to the end of the default value of the **log.dirs** configuration item. Use commas (,) to separate multiple directories. For example:

${BIGDATA_DATA_HOME}/kafka/data1/kafka-logs,${BIGDATA_DATA_HOME}/kafka/data2/kafka-logs

Step 4 Click **Save**, and then click **OK**. When **Operation succeeded** is displayed, click **Finish**.

Step 5 Choose **Cluster** > **Services** > **Kafka**. In the upper right corner, choose **More** > **Restart Service** to restart the Kafka service.

Changing the storage directory of a single Kafka instance

Step 6 Log in to the Broker node as user **root** and perform the following operations:

1. Create a target directory.

   For example, to create the target directory **${BIGDATA_DATA_HOME}/kafka/data2**, run the following command:

   mkdir ${BIGDATA_DATA_HOME}/kafka/data2

2. Mount the directory to the new disk. For example, mount **${BIGDATA_DATA_HOME}/kafka/data2** to the new disk.
3. Modify permissions on the new directory.

   For example, to modify permissions on the **${BIGDATA_DATA_HOME}/kafka/data2** directory, run the following commands:

   chmod 700 ${BIGDATA_DATA_HOME}/kafka/data2 -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/kafka/data2 -R**

Step 7 Log in to FusionInsight Manager for MRS 3.*x* or later, and choose **Cluster** > **Services** > **Kafka** > **Instance**.

Step 8 Click the specified broker instance and switch to **Instance Configurations**.

Enter **log.dirs** in the search box and add the new directory to the end of the default value of the **log.dirs** configuration item. Use commas (,) to separate

multiple directories, for example, **${BIGDATA_DATA_HOME}/kafka/data1/kafka-logs,${BIGDATA_DATA_HOME}/kafka/data2/kafka-logs**.

Step 9 Click **Save**, and then click **OK**. A message is displayed, indicating that the operation is successful. Click **Finish**.

Step 10 On the Broker instance page, choose **More** > **Restart Instance** to restart the Broker instance.

----End

12.14.14 Checking the Consumption Status of Consumer Group

Scenario

This section describes how to view the current expenditure on the client based on service requirements.

This section applies to MRS 3.*x* or later.

Prerequisites

- The system administrator has understood service requirements and prepared a system user.
- The Kafka client has been installed.

Procedure

Step 1 Log in as a client installation user to the node on which the Kafka client is installed.

Step 2 Switch to the Kafka client installation directory, for example, **/opt/kafkaclient**.

cd /opt/kafkaclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to perform user authentication (skip this step in normal mode):

kinit *Component service user*

Step 5 Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

Step 6 Run the **kafka-consumer-groups.sh** command to check the current consumption status.

- Check the Consumer Group list on Kafka saved by Offset:

  ./kafka-consumer-groups.sh --list --bootstrap-server <*Service IP address of any broker node:21007*> **--command-config ../config/consumer.properties**

  eg:./kafka-consumer-groups.sh --bootstrap-server 192.168.1.1:21007 --list --command-config ../config/consumer.properties

- Check the consumption status of Consumer Group on Kafka saved by Offset:

  ./kafka-consumer-groups.sh --describe --bootstrap-server <*Service IP address of any broker node:21007*> *--group Consumer group name* **--command-config ../config/consumer.properties**

  eg:./kafka-consumer-groups.sh --describe --bootstrap-server 192.168.1.1:21007 --group example-group --command-config ../config/consumer.properties

NOTICE

1. Ensure that the current consumer is online and consumes data.
2. Configure the **group.id** in the **consumer.properties** configuration file and **--group** in the command to the group to be queried.
3. The Kafka cluster's IP port number is 21007 in security mode and 9092 in normal mode.

----End

12.14.15 Kafka Balancing Tool Instructions

Scenario

This section describes how to use the Kafka balancing tool on a client to balance the load of the Kafka cluster based on service requirements in scenarios such as node decommissioning, node recommissioning, and load balancing.

This section applies to MRS 3.*x* or later. For versions earlier than MRS 3.*x*, see **Balancing Data After Kafka Node Scale-Out**.

Prerequisites

- The system administrator has understood service requirements and prepared a Kafka administrator (belonging to the **kafkaadmin** group. It is not required for the normal mode.).
- The Kafka client has been installed.

Procedure

Step 1 Log in as a client installation user to the node on which the Kafka client is installed.

Step 2 Switch to the Kafka client installation directory, for example, **/opt/kafkaclient**.

cd /opt/kafkaclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to authenticate the user (skip this step in normal mode):

kinit *Component service user*

Step 5 Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka

Step 6 Run the **kafka-balancer.sh** command to balance user cluster. The commonly used commands are:

- Run the **--run** command to perform cluster balancing:

  ./bin/kafka-balancer.sh --run --zookeeper *<ZooKeeper service IP address of any ZooKeeper node:zkPort/**kafka**>* **--bootstrap-server** *<Kafka cluster IP: port>* **--throttle 10000000 --consumer-config config/consumer.properties --enable-az-aware --show-details**

  This command consists of generation and execution of the balancing solution. **--show-details** is optional, indicating whether to print the solution details. **--throttle** indicates the bandwidth limit during the execution of the balancing solution. The unit is bytes per second (bytes/sec). **--enable-az-aware** indicates that the cross-AZ feature is enabled when the balancing solution is generated. When this parameter is used, ensure that the cross-AZ feature has been enabled for the cluster.

- Run the **--run** command to decommission a node:

  ./bin/kafka-balancer.sh --run --zookeeper *<Service IP address of any ZooKeeper node:zkPort/**kafka**>* **--bootstrap-server** *<Kafka cluster IP address: port>* **--throttle 10000000 --consumer-config config/consumer.properties --remove-brokers** *<BrokerId list>* **--enable-az-aware --force**

  In the command, **--remove-brokers** indicates the list of broker IDs to be deleted. Multiple broker IDs are separated by commas (,). **--force** is optional, indicating that the disk usage alarm is ignored and the migration solution is forcibly generated. **-enable-az-aware** is optional, indicating that the cross-AZ feature is enabled when the balancing solution is generated. When this parameter is used, ensure that the cross-AZ feature has been enabled for the cluster.

- Run the following command to view the execution status:

  ./bin/kafka-balancer.sh --status --zookeeper *<Service IP address of any ZooKeeper node:zkPort/**kafka**>*

- Run the following command to generate a balancing solution:

  ./bin/kafka-balancer.sh --generate --zookeeper *<Service IP address of any ZooKeeper node:zkPort/**kafka**>* **--bootstrap-server** *<Kafka cluster IP address:port>* **--consumer-config config/consumer.properties --enable-az-aware**

  This command is used to generate a migration solution based on the current cluster status and print the solution to the console. **--enable-az-aware** is optional, indicating that the cross-AZ feature is enabled when a migration solution is generated. If this parameter is used, ensure that the cross-AZ feature has been enabled for the cluster.

- Clearing the intermediate status

  ./bin/kafka-balancer.sh --clean --zookeeper *<Service IP address of any ZooKeeper node:zkPort/**kafka**>*

  This command is used to clear the intermediate status information on the ZooKeeper when the migration is not complete.

NOTICE

The port number of the Kafka cluster's IP address is 21007 in security mode and 9092 in normal mode.

----End

Troubleshooting

During partition migration using the Kafka balancing tool, if the execution progress of the balancing tool is blocked due to a Broker fault in the cluster, you need to manually rectify the fault. The scenarios are as follows:

- The Broker is faulty because the disk usage reaches 100%.
  a. Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance**, stop the Broker instance in the **Restoring** state, and record the management IP address of the node where the instance resides and the corresponding **broker.id**. You can click the role name to view the value, on the **Instance Configurations** page, select **All Configurations** and search for the **broker.id** parameter.
  b. Log in to the recorded management IP address as user **root**, and run the **df -lh** command to view the mounted directory whose disk usage is 100%, for example, **${BIGDATA_DATA_HOME}/kafka/data1**.
  c. Go to the directory, run the **du -sh *** command to view the size of each file in the directory, Check whether files other than files in the **kafka-logs** directory exist, and determine whether these files can be deleted or migrated.
     - If yes, delete or migrate the related data and go to **8**.
     - If no, go to **4**.
  d. Go to the **kafka-logs** directory, run the **du -sh *** command, select a partition folder to be moved. The naming rule is **Topic name-Partition ID**. Record the topic and partition.
  e. Modify the **recovery-point-offset-checkpoint** and **replication-offset-checkpoint** files in the **kafka-logs** directory in the same way.
     i. Decrease the number in the second line in the file. (To remove multiple directories, the number deducted is equal to the number of files to be removed.
     ii. Delete the line of the to-be-removed partition. (The line structure is "*Topic name Partition ID Offset*". Save the data before deletion. Subsequently, the content must be added to the file of the same name in the destination directory.)
  f. Modify the **recovery-point-offset-checkpoint** and **replication-offset-checkpoint** files in the destination data directory (for example, **${BIGDATA_DATA_HOME}/kafka/data2/kafka-logs**) in the same way.
     - Increase the number in the second line in the file. (To move multiple directories, the number added is equal to the number of files to be moved.

- Add the to-be moved partition to the end of the file. (The line structure is "*Topic name Partition ID Offset*". You can copy the line data saved in **5**.)

g. Move the partition to the destination directory. After the partition is moved, run the **chown omm:wheel -R** *Partition directory* command to modify the directory owner group for the partition.

h. Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Kafka** > **Instance** to start the stopped Broker instance.

i. Wait for 5 to 10 minutes and check whether the health status of the Broker instance is **Good**.

- If yes, resolve the disk capacity insufficiency problem according to the handling method of "ALM-38001 Insufficient Kafka Disk Capacity" after the alarm is cleared.
- If no, contact O&M support.

After the faulty Broker is recovered, the blocked balancing task continues. You can run the **--status** command to view the task execution progress.

- The Broker fault occurs because of other causes, the fault scenario is clear, and the fault can be rectified within a short period of time.

a. Restore the faulty Broker according to the root cause.

b. After the faulty Broker is recovered, the blocked balancing task continues. You can run the **--status** command to view the task execution progress.

- The Broker fault occurs because of other causes, the fault scenario is complex, and the fault cannot be rectified within a short period of time.

a. Run the **kinit** *Kafka administrator account* command (skip this step in normal mode).

b. Run the **zkCli.sh -server <***ZooKeeper cluster service IP address:zkPort***/kafka>** command to log in to ZooKeeper Shell.

c. Run the **addauth krbgroup** command (skip this step in normal mode).

d. Delete the **/admin/reassign_partitions** and **/controller** directories.

e. Perform the preceding steps to forcibly stop the migration. After the cluster recovers, run the **kafka-reassign-partitions.sh** command to delete redundant copies generated during the intermediate process.

12.14.16 Balancing Data After Kafka Node Scale-Out

Scenario

This section describes how to use the Kafka balancing tool on the client to balance the load of the Kafka cluster after Kafka nodes are scaled out.

This section applies to versions earlier than MRS 3.*x*. For MRS 3.*x* or later, see **Kafka Balancing Tool Instructions**.

Prerequisites

- The system administrator has understood service requirements and prepared a Kafka administrator (belonging to the **kafkaadmin** group and not required for the normal mode).
- The Kafka client has been installed, for example, in the **/opt/kafkaclient** directory.
- Two topics named **test_2** and **test_3** has been created by referring to **Step 7**. The **move-kafka-topic.json** file has been created in the **/opt/kafkaclient/Kafka/kafka** directory. The topic format is as follows:

```
{
"topics":
[{"topic":"test_2"},{"topic":"test_3"}],
"version":1
}
```

Procedure

Step 1 Log in to the node where the Kafka client is installed as the client installation user.

Step 2 Run the following command to switch to the client installation directory:

cd /opt/kafkaclient

Step 3 Run the following command to set environment variables:

source bigdata_env

Step 4 Run the following command to perform user authentication (skip this step if the cluster is in normal mode):

kinit *Component service user*

Step 5 Run the following command to go to the **bin** directory of the Kafka client:

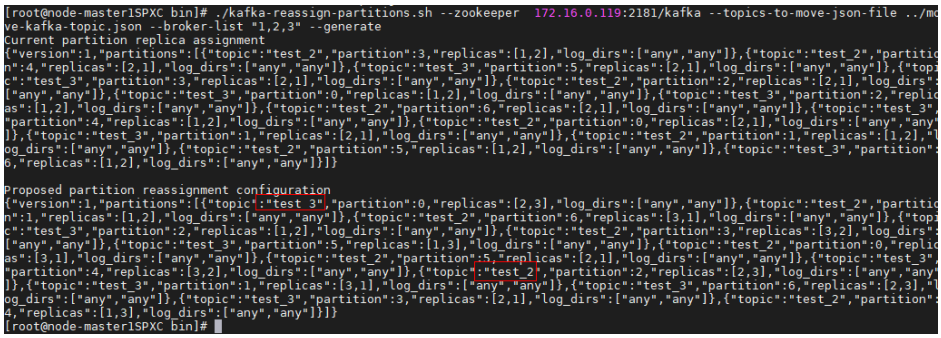
cd Kafka/kafka/bin

Step 6 Run the following command to generate an execution plan:

./kafka-reassign-partitions.sh --zookeeper *172.16.0.119:***2181/kafka --topics-to-move-json-file ../move-kafka-topic.json --broker-list "***1,2,3***" --generate**

NOTE

- **172.16.0.119**: service IP address of the ZooKeeper instance
- **--broker-list "1,2,3"**: list of broker instances. **1,2,3** indicates all broker IDs after a scale-out.



Step 7 Run the **vim ../reassignment.json** command to create the **reassignment.json** file and save it to the **/opt/kafkaclient/Kafka/kafka** directory.

Copy the content under **Proposed partition reassignment configuration** generated in **Step 6** to the **reassignment.json** file, as shown in the follows:

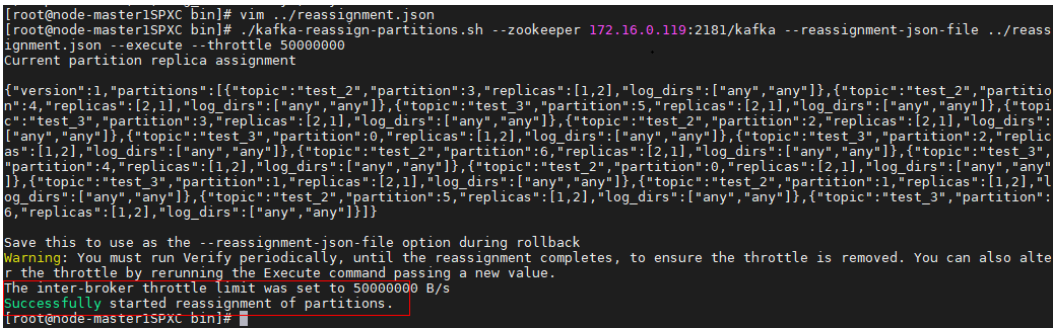
```
{"version":1,"partitions":[{"topic":"test","partition":4,"replicas":[1,2],"log_dirs":["any","any"]},
{"topic":"test","partition":1,"replicas":[1,3],"log_dirs":["any","any"]},{"topic":"test","partition":3,"replicas":
[3,1],"log_dirs":["any","any"]},{"topic":"test","partition":0,"replicas":[3,2],"log_dirs":["any","any"]},
{"topic":"test","partition":2,"replicas":[2,1],"log_dirs":["any","any"]}]}
```

Step 8 Run the following command to redistribute partitions:

./kafka-reassign-partitions.sh --zookeeper *172.16.0.119:***2181/kafka --reassignment-json-file ../reassignment.json --execute --throttle** *50000000*

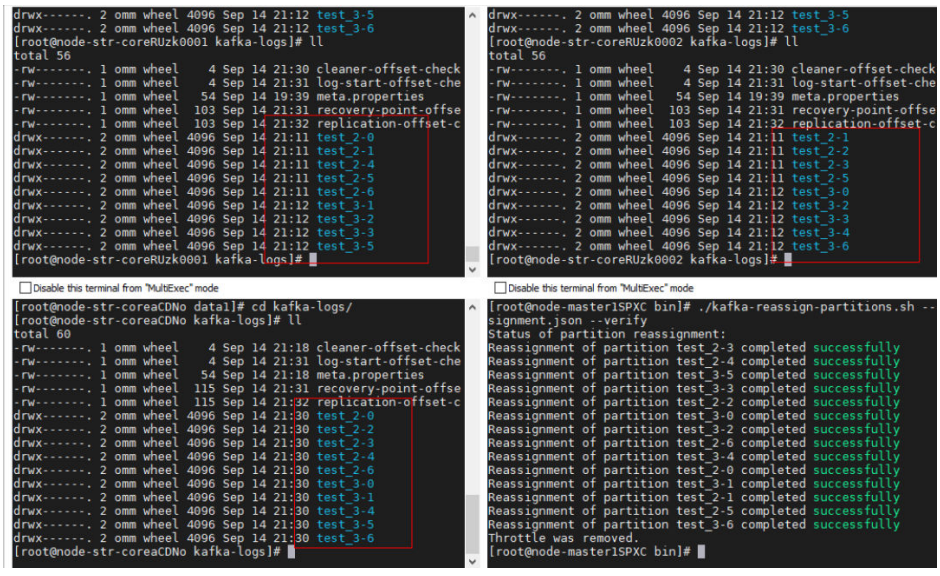
NOTE

--throttle 50000000: The maximum bandwidth is 50 MB/s. You can change the bandwidth based on the data volume and the customer's requirements on the balancing time. If the data volume is 5 TB, the bandwidth is 50 MB/s and the data balancing takes about 8 hours.



Step 9 Run the following command to check the data migration status:

./kafka-reassign-partitions.sh --zookeeper *172.16.0.119:***2181/kafka --reassignment-json-file ../reassignment.json --verify**



----End

12.14.17 Kafka Token Authentication Mechanism Tool Usage

Scenario

Operations need to be performed on tokens when the token authentication mechanism is used.

This section applies to security clusters of MRS 3.*x* or later.

Prerequisites

- The system administrator has understood service requirements and prepared a system user.
- The Kafka client has been installed.

Procedure

Step 1 Log in as a client installation user to the node on which the Kafka client is installed.

Step 2 Switch to the Kafka client installation directory, for example, **/opt/kafkaclient**.

cd /opt/kafkaclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to perform user authentication:

kinit *Component service user*

Step 5 Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

Step 6 Use **kafka-delegation-tokens.sh** to perform operations on tokens.

- Generate a token for a user.

  ./kafka-delegation-tokens.sh --create --bootstrap-server <*IP1:PORT, IP2:PORT,...***> --max-life-time-period <***Long: max life period in milliseconds***> --command-config <***config file***> --renewer-principal User:<***user name***>**

  Example: **./kafka-delegation-tokens.sh --create --bootstrap-server 192.168.1.1:21007,192.168.1.2:21007,192.168.1.3:21007 --command-config ../config/producer.properties --max-life-time-period -1 --renewer-principal User:username**

- List information about all tokens of a specified user.

  ./kafka-delegation-tokens.sh --describe --bootstrap-server <*IP1:PORT, IP2:PORT,...***> --command-config <***config file***> --owner-principal User:<***user name***>**

  Example: **./kafka-delegation-tokens.sh --describe --bootstrap-server 192.168.1.1:21007,192.168.1.2:21007,192.168.1.3:21007 --command-config ../config/producer.properties --owner-principal User:username**

- Update the token validity period.

./kafka-delegation-tokens.sh --renew --bootstrap-server <*IP1:PORT, IP2:PORT,...***> --renew-time-period <***Long: renew time period in milliseconds***> --command-config <***config file***> --hmac <***String: HMAC of the delegation token***>**

Example: **./kafka-delegation-tokens.sh --renew --bootstrap-server 192.168.1.1:21007,192.168.1.2:21007,192.168.1.3:21007 --renew-time-period -1 --command-config ../config/producer.properties --hmac ABCDEFG**

- Destroy a token.

  ./kafka-delegation-tokens.sh --expire --bootstrap-server <*IP1:PORT, IP2:PORT,...***> --expiry-time-period <***Long: expiry time period in milliseconds***> --command-config <***config file***> --hmac <***String: HMAC of the delegation token***>**

  Example: **./kafka-delegation-tokens.sh --expire --bootstrap-server 192.168.1.1:21007,192.168.1.2:21007,192.168.1.3:21007 --expiry-time-period -1 --command-config ../config/producer.properties --hmac ABCDEFG**

----End

12.14.18 Introduction to Kafka Logs

This section applies to MRS 3.*x* or later.

Log Description

Log paths: The default storage path of Kafka logs is **/var/log/Bigdata/kafka**. The default storage path of audit logs is **/var/log/Bigdata/audit/kafka**.

- Broker: **/var/log/Bigdata/kafka/broker** (run logs)

Log archive rule: The automatic Kafka log compression function is enabled. By default, when the size of logs exceeds 30 MB, logs are automatically compressed into a log file named in the following format: *<Original log file name>-<yyyy-mm-dd_hh-mm-ss>.[ID]*.**log.zip**. A maximum of 20 latest compressed files are retained by default. You can configure the number of compressed files and the compression threshold.

Table 12-269 Broker log list

Type	Log File Name	Description
Run log	server.log	Server run log of the broker process
	controller.log	Controller run log of the broker process
	kafka-request.log	Request run log of the broker process
	log-cleaner.log	Cleaner run log of the broker process

Type	Log File Name	Description
	state-change.log	State-change run log of the broker process
	kafkaServer-<SSH_USER>-<DATE>-<PID>-gc.log	GC log of the broker process
	postinstall.log	Work log after broker installation
	prestart.log	Work log before broker startup
	checkService.log	Log that records whether broker starts successfully
	start.log	Startup log of the broker process
	stop.log	Stop log of the broker process
	checkavailable.log	Log that records the health check details of the Kafka service
	checkInstanceHealth.log	Log that records the health check details of broker instances
	kafka-authorizer.log	Broker authorization log
	kafka-root.log	Broker basic log
	cleanup.log	Cleanup log of broker uninstallation
	metadata-backup-recovery.log	Broker backup and recovery log
	ranger-kafka-plugin-enable.log	Log that records the Ranger plug-ins enabled by brokers
	server.out	Broker JVM log
	audit.log	Authentication log of the Ranger authentication plug-in. This log is archived in the **/var/log/Bigdata/audit/kafka** directory.

Log Level

Table 12-270 describes the log levels supported by Kafka.

Levels of run logs are ERROR, WARN, INFO, and DEBUG from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-270 Log levels

Level	Description
ERROR	Logs of this level record error information about system running.
WARN	Logs of this level record exception information about the current event processing.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page. See **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

----End

Log Format

The following table describes the Kafka log format.

Table 12-271 Log formats

Type	Format	Example
Run log	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<Thread that generates the log>| <Message in the log>| <Full name of the log event invocation class>(<Log file>:<Row>)	2015-08-08 11:09:53,483 | INFO | [main] | Loading logs. | kafka.log.LogManager (Logging.scala:68)

Type	Format	Example
	<yyyy-MM-dd HH:mm:ss><HostName> <Component name><logLevel><Message>	2015-08-08 11:09:51 10-165-0-83 Kafka INFO Running kafka-start.sh.

12.14.19 Performance Tuning

12.14.19.1 Kafka Performance Tuning

Scenario

You can modify Kafka server parameters to improve Kafka processing capabilities in specific service scenarios.

Parameter Tuning

Modify the service configuration parameters. For details, see **Modifying Cluster Service Configuration Parameters**. For details about the tuning parameters, see **Table 12-272**.

Table 12-272 Tuning parameters

Parameter	Default Value	Scenario
num.recovery.threads.per.data.dir	10	During the Kafka startup process, if a large volume of data exists, you can increase the value of this parameter to accelerate the startup.
background.threads	10	Specifies the number of threads processed by a broker background task. If a large volume of data exists, you can increase the value of this parameter to improve broker processing capabilities.
num.replica.fetchers	1	Specifies the number of threads used when a replica requests to the Leader for data synchronization. If the value of this parameter is increased, the replica I/O concurrency increases.

Parameter	Default Value	Scenario
num.io.threads	8	Specifies the number of threads used by the broker to process disk I/O. It is recommended that the number of threads be greater than or equal to the number of disks.
KAFKA_HEAP_OPTS	-Xmx6G -Xms6G	Specifies the Kafka JVM heap memory setting. If the data volume on the broker is large, adjust the heap memory size.

12.14.20 Kafka Feature Description

Kafka Idempotent Feature

Feature description: The function of creating idempotent producers is introduced in Kafka 0.11.0.0. After this function is enabled, producers are automatically upgraded to idempotent producers. When producers send messages with the same field values, brokers automatically detect whether the messages are duplicate to avoid duplicate data. Note that this feature can only ensure idempotence in a single partition. That is, an idempotent producer can ensure that no duplicate messages exist in a partition of a topic. Only idempotence on a single session can be implemented. The session refers to the running of the producer process. That is, idempotence cannot be ensured after the producer process is restarted.

Method for enabling this feature:

1. Add **props.put("enable.idempotence", true)** to the secondary development code.
2. Add **enable.idempotence = true** to the client configuration file.

Kafka Transaction Feature

Feature description: Kafka 0.11 introduces the transaction feature. The Kafka transaction feature indicates that a series of producer message production and consumer offset submission operations are in the same transaction, or are regarded as an atomic operation. Message production and offset submission succeed or fail at the same time. This feature provides transactions at the Read Committed isolation level to ensure that multiple messages are written to the target partition atomically and that the consumer can view only the transaction messages that are successfully submitted. The transaction feature of Kafka is used in the following scenarios:

1. Multiple pieces of data sent by a producer can be encapsulated in a transaction to form an atomic operation. All messages are successfully sent or fail to be sent.
2. read-process-write mode: Message consumption and production are encapsulated in a transaction to form an atomic operation. In a streaming

application, a service usually needs to receive messages from the upstream system, process the messages, and then send the processed messages to the downstream system. This corresponds to message consumption and production.

Example of secondary development code:

```
// Initialize the configuration and enable the transaction feature.
Properties props = new Properties();
props.put("enable.idempotence", true);
props.put("transactional.id", "transaction1");
...

KafkaProducer producer = new KafkaProducer<String, String>(props);

// init transaction
producer.initTransactions();
try {
    // Start a transaction.
    producer.beginTransaction();
    producer.send(record1);
    producer.send(record2);
    // Stop a transaction.
    producer.commitTransaction();
} catch (KafkaException e) {
    // Abort a transaction.
    producer.abortTransaction();
}
```

Nearby Consumption

Feature description: In versions earlier than Kafka 2.4.0, the production and consumption of the client are leader copies oriented to each partition. Follower copies are used only for data redundancy and do not provide services for external systems. As a result, the leader copy has high pressure. In addition, in cross-DC and cross-rack consumption scenarios, a large volume of data is transmitted between DCs and between racks. In Kafka 2.4.0 and later versions, the Kafka kernel can consume data from follower replicas, which greatly reduces the data transmission volume and reduces the network bandwidth pressure in cross-DC and cross-rack scenarios. The community opens the ReplicaSelector API to support this feature. By default, MRS Kafka provides two methods to use this API.

1. **RackAwareReplicaSelector**: indicates that replicas in the same rack are preferentially consumed (nearby consumption in a rack).
2. **AzAwareReplicaSelector**: indicates that copies from nodes in the same AZ are preferentially consumed (nearby consumption in an AZ).

The following uses **RackAwareReplicaSelector** as an example to describe how to consume the closest replica.

```
public class RackAwareReplicaSelector implements ReplicaSelector {

  @Override
  public Optional<ReplicaView> select(TopicPartition topicPartition,
                                      ClientMetadata clientMetadata,
                                      PartitionView partitionView) {
    if (clientMetadata.rackId() != null && !clientMetadata.rackId().isEmpty()) {
      Set<ReplicaView> sameRackReplicas = partitionView.replicas().stream()
              // Filter the replicas that are in the same rack as the client.
              .filter(replicaInfo -> clientMetadata.rackId().equals(replicaInfo.endpoint().rack()))
              .collect(Collectors.toSet());
      if (sameRackReplicas.isEmpty()) {
        // If no replicas are in the same rack as the client, the leader replica is returned.
        return Optional.of(partitionView.leader());
```

```
            } else {
                // It shows that a replica that is in the same rack as the client exists.
                if (sameRackReplicas.contains(partitionView.leader())) {
                    // If the client and the leader replica are in the same rack, the leader replica returns first.
                    return Optional.of(partitionView.leader());
                } else {
                    // Otherwise, the latest replica synchronized with the leader is returned.
                    return sameRackReplicas.stream().max(ReplicaView.comparator());
                }
            }
        } else {
            // If the rack information is not contained in the client request, the leader replica is returned first.
            return Optional.of(partitionView.leader());
        }
    }
}
```

Method for enabling this feature:

1. Server: Update the **replica.selector.class** configuration item based on different features.
   - To enable "nearby consumption in a rack", set this parameter to **org.apache.kafka.common.replica.RackAwareReplicaSelector**.
   - To enable "nearby consumption in an AZ", set this parameter to **org.apache.kafka.common.replica.AzAwareReplicaSelector**.
2. Client: Add the **client.rack** configuration item to the **consumer.properties** file in the *{Client installation directory}***/Kafka/kafka/config** directory.
   - If the "nearby consumption in a rack" is enabled on the server, add the information about the rack where the client is located, for example, **client.rack = /default0/rack1**.
   - If the "nearby consumption in an AZ" is enabled on the server, add the information about the rack where the client is located, for example, **client.rack = /AZ1/rack1**.

Ranger Unified Authentication

Feature description: In versions earlier than Kafka 2.4.0, Kafka supports only the SimpleAclAuthorizer authentication plugin provided by the community. In Kafka 2.4.0 and later versions, MRS Kafka supports both the Ranger authentication plugin and the authentication plugin provided by the community. Ranger authentication is used by default. Based on the Ranger authentication plugin, fine-grained Kafka ACL management can be performed.

NOTE

If the Ranger authentication plugin is used on the server and **allow.everyone.if.no.acl.found** is set to **true**, all actions are allowed when a non-secure port is used for access. You are advised to disable **allow.everyone.if.no.acl.found** for security clusters that use the Ranger authentication plugin.

12.14.21 Migrating Data Between Kafka Nodes

Scenario

This section describes how to use Kafka client commands to migrate partition data between disks on a node without stopping the Kafka service.

Prerequisites

- The system administrator has understood service requirements and prepared a Kafka user (belonging to the **kafkaadmin** group and not required for the normal mode).
- The Kafka client has been installed.
- The Kafka instance status and disk status are normal.
- Based on the current disk space usage of the partition to be migrated, ensure that the disk space will be sufficient after the migration.

Procedure

Step 1 Log in as a client installation user to the node on which the Kafka client is installed.

Step 2 Run the following command to switch to the Kafka client installation directory, for example, **/opt/kafkaclient**:

cd /opt/kafkaclient

Step 3 Run the following command to set environment variables:

source bigdata_env

Step 4 Run the following command to authenticate the user (skip this step in normal mode):

kinit *Component service user*

Step 5 Run the following command to switch to the Kafka client directory:

cd Kafka/kafka/bin

Step 6 Run the following command to view the topic details of the partition to be migrated:

Security mode:

./kafka-topics.sh --describe --bootstrap-server *IP address of the Kafkacluster:21007* **--command-config ../config/client.properties --topic** *topic name*

Normal mode:

./kafka-topics.sh --describe --bootstrap-server *IP address of the Kafka cluster:21005* **--command-config ../config/client.properties --topic** *Topic name*

Step 7 Run the following command to query the mapping between **Broker_ID** and the IP address:

./kafka-broker-info.sh --zookeeper *IP address of the ZooKeeper quorumpeer instance*:*ZooKeeper port number*/**kafka**

```
Broker_ID     IP_Address
--------------------------
4             192.168.0.100
5             192.168.0.101
6             192.168.0.102
```

NOTE

- IP address of the ZooKeeper quorumpeer instance

  To obtain IP addresses of all ZooKeeper quorumpeer instances, log in to FusionInsight Manager and choose **Cluster** > **Services** > **ZooKeeper**. On the displayed page, click **Instance** and view the IP addresses of all the hosts where the quorumpeer instances locate.

- Port number of the ZooKeeper client

  Log in to FusionInsight Manager and choose **Cluster** > **Service** > **ZooKeeper**. On the displayed page, click **Configurations** and check the value of **clientPort**. The default value is **24002**.

Step 8 Obtain the partition distribution and node information from the command output in **Step 6** and **Step 7**, and create the JSON file for reallocation in the current directory.

To migrate data in the partition whose **Broker_ID** is **6** to the **/srv/BigData/hadoop/data1/kafka-logs** directory, the required JSON configuration file is as follows:

```
{"partitions":[{"topic": "testws","partition": 2,"replicas": [6,5],"log_dirs": ["/srv/BigData/hadoop/data1/kafka-logs","any"]}],"version":1}
```

NOTE

- **topic** indicates the topic name, for example, **testws**.
- **partition** indicates the topic partition.
- The number in **replicas** corresponds to **Broker_ID**.
- **log_dirs** indicates the path of the disk to be migrated. In this example, **log_dirs** of the node whose **Broker_ID** is **5** is set to **any**, and that of the node whose **Broker_ID** is **6** is set to **/srv/BigData/hadoop/data1/kafka-logs.** Note that the path must correspond to the node.

Step 9 Run the following command to perform reallocation:

Security mode:

./kafka-reassign-partitions.sh --bootstrap-server *Service IP address of Broker*:**21007 --command-config ../config/client.properties --zookeeper** *{zk_host}:{port}*/**kafka --reassignment-json-file** *Path of the JSON file compiled in* ***Step 8*** **--execute**

Normal mode:

./kafka-reassign-partitions.sh --bootstrap-server *Service IP address of Broker*:**21005 --command-config ../config/client.properties --zookeeper** *{zk_host}:{port}*/**kafka --reassignment-json-file** *Path of the JSON file compiled in* ***Step 8*** **--execute**

If message "Successfully started reassignment of partitions" is displayed, the execution is successful.

----End

12.14.22 Common Issues About Kafka

12.14.22.1 How Do I Solve the Problem that Kafka Topics Cannot Be Deleted?

Question

How do I delete a Kafka topic if it fails to be deleted?

Answer

- Possible cause 1: The **delete.topic.enable** configuration item is not set to **true**. The deletion can be performed only when the configuration item is set to **true**.
- Possible cause 2: The **auto.create.topics.enable** configuration parameter is set to **true**, which is used by other applications and is always running in the background.

Solution:

- For cause 1: Set **delete.topic.enable** to **true** on the configuration page.
- For cause 2: Stop the application that uses the topic in the background, or set **auto.create.topics.enable** to **false** (restart the Kafka service), and then delete the topic.

12.15 Using KafkaManager

12.15.1 Introduction to KafkaManager

KafkaManager is a tool for managing Apache Kafka and provides GUI-based metric monitoring and management of Kafka clusters.

KafkaManager supports the following functions:

- Manage multiple Kafka clusters.
- Check cluster status (topics, consumers, offsets, partitions, replicas, and nodes)
- Run preferred replica election.
- Generate partition assignments with option to select brokers to use.
- Run reassignment of partitions (based on generated assignments).
- Create a topic with optional topic configurations (Multiple Kafka cluster versions are supported).
- Delete a topic (only supported on 0.8.2+ and **delete.topic.enable = true** is set in broker configuration).

- Batch generate partition assignments for multiple topics with option to select brokers to use.
- Batch run reassignment of partitions for multiple topics.
- Add partitions to an existing topic.
- Update configurations for an existing topic.
- Optionally enable JMX polling for broker-level and topic-level metrics.
- Optionally filter out consumers that do not have ids/owner/&offsets/ directories in ZooKeeper.

12.15.2 Accessing the KafkaManager Web UI

You can monitor and manage Kafka clusters on the graphical KafkaManager web UI.

Prerequisites

- KafkaManager has been installed in a cluster.
- The password of user **admin** has been obtained. The password of user **admin** is specified by the user during MRS cluster creation.

Accessing the KafkaManager Web UI

Step 1 In the **KafkaManager Summary** area, click any UI link in **KafkaManager WebUI** to access the KafkaManager web UI.

You can view the following information on the KafkaManager web UI.

- Kafka cluster list
- Broker node list and metric monitoring information of Kafka clusters
- Kafka cluster replica monitoring information
- Kafka cluster consumer monitoring information

NOTE

You can click the KafkaManager logo in the upper left corner on any sub-page of KafkaManager to return to the homepage of the KafkaManager web UI, where a cluster list is displayed.

----End

12.15.3 Managing Kafka Clusters

Kafka cluster management includes the following operations:

- **Adding a Cluster on the KafkaManager Web UI**
- **Updating Cluster Parameters**
- **Deleting a Cluster on the KafkaManager Web UI**

Adding a Cluster on the KafkaManager Web UI

After a Kafka cluster is created for the first time, a default Kafka cluster named **my-cluster** is created on the KafkaManager web UI. You can also add Kafka

clusters that have been created on the MRS management console on the KafkaManager web UI to manage multiple Kafka clusters.

Step 1 Log in to the KafkaManager web UI.

Step 2 In the upper part of the page, choose **Cluster** > **Add Cluster**.

Step 3 Set the cluster parameters. For the following parameters, refer to their example values. Retain the default values for other parameters.

Table 12-273 Cluster parameters to be modified

Parameter	Example Value	Description
Cluster Name	mrs-demo	Name of the cluster to be added on the KafkaManager web UI
Cluster Zookeeper Hosts	zk1_ip:zk1_port, zk2_ip:zk2_port/kafka	ZooKeeper address of the cluster to be added
Kafka Version	1.1.0	Kafka version of the cluster to be added. The default value is **1.1.0**.
Enable JMX Polling (Set JMX_PORT env variable before starting kafka server)	Selected	-
Poll consumer information (Not recommended for large # of consumers)	Selected	-
Enable Active OffsetCache (Not recommended for large # of consumers)	Selected	-
Display Broker and Topic Size (only works after applying this patch)	Selected	-
Security Protocol	PLAINTEXT	• For a Kafka cluster with Kerberos authentication enabled, select **SASL_PLAINTEXT**. • For a Kafka cluster with Kerberos authentication disabled, select **PLAINTEXT**.

Step 4 Click **Save**.

----End

Updating Cluster Parameters

Step 1 Log in to the KafkaManager web UI.

Step 2 Click **Modify** in the **Operations** column of the cluster.

Step 3 Go to the cluster configuration page and modify cluster parameters.

----End

Deleting a Cluster on the KafkaManager Web UI

Step 1 Log in to the KafkaManager web UI.

Step 2 Click **Disable** in the **Operations** column of the cluster.

Step 3 When **Delete** or **Enable** is displayed in the **Operations** column on the cluster list page, click **Delete** to delete the cluster. You can also click **Enable** to enable the cluster.

----End

12.15.4 Kafka Cluster Monitoring Management

The Kafka cluster monitoring management includes the following operations:

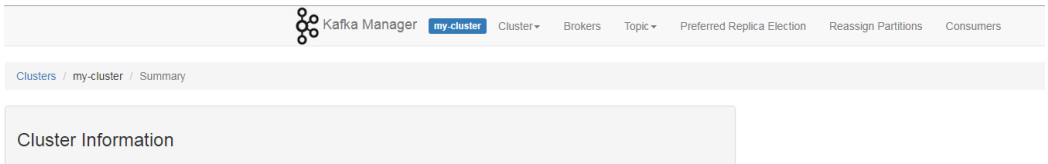
- **Viewing Broker Information**
- **Viewing Topic Information**
- **Viewing Consumers Information**
- **Modifying the Partition of a Topic Through KafkaManager**

Viewing Broker Information

Step 1 Log in to the KafkaManager web UI.

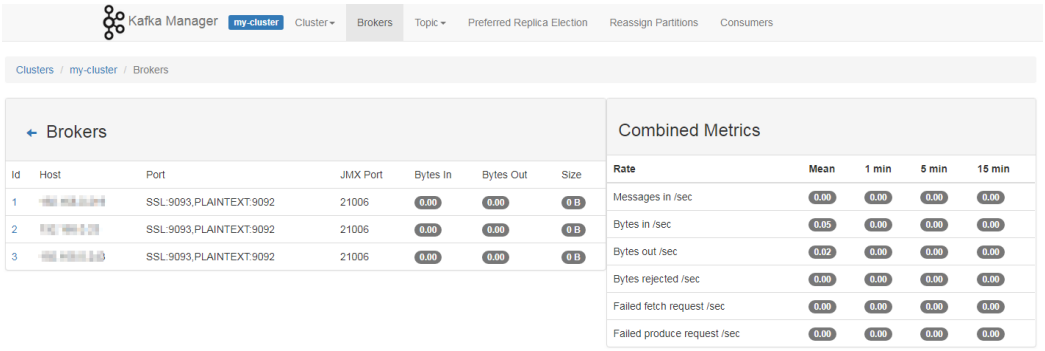
Step 2 On the cluster list page, click a cluster name to access the Summary page of the cluster.

Figure 12-26 Summary page of a cluster



Step 3 Click **Brokers** to access the Broker monitoring page. The page displays the Broker list and I/O statistics of the Broker nodes.

Figure 12-27 Broker monitoring page



----End

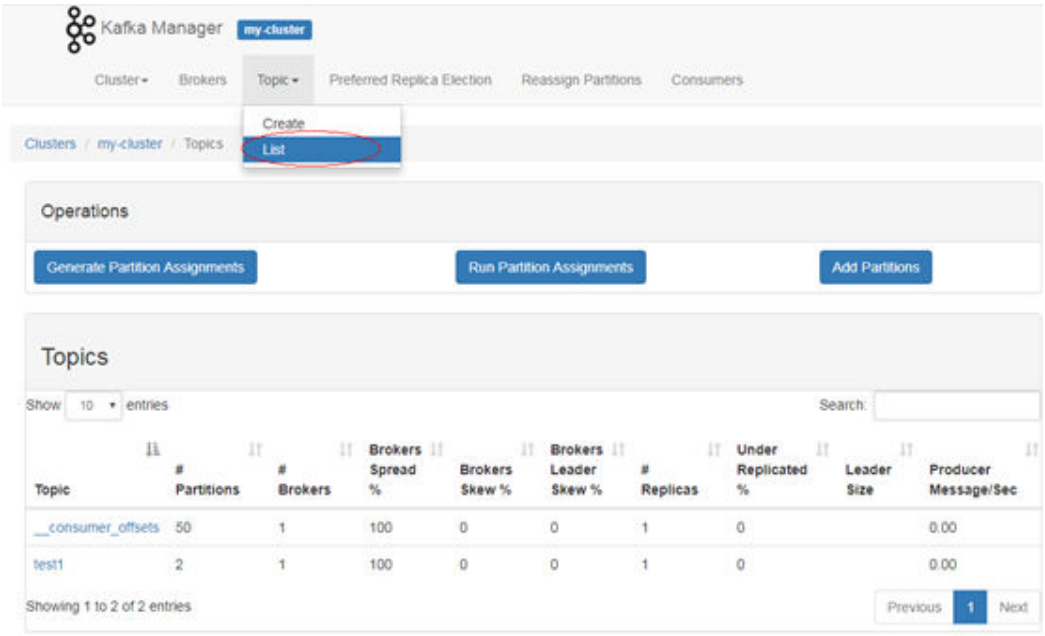
Viewing Topic Information

Step 1 Log in to the KafkaManager web UI.

Step 2 On the cluster list page, click a cluster name to access the **Summary** page of the cluster.

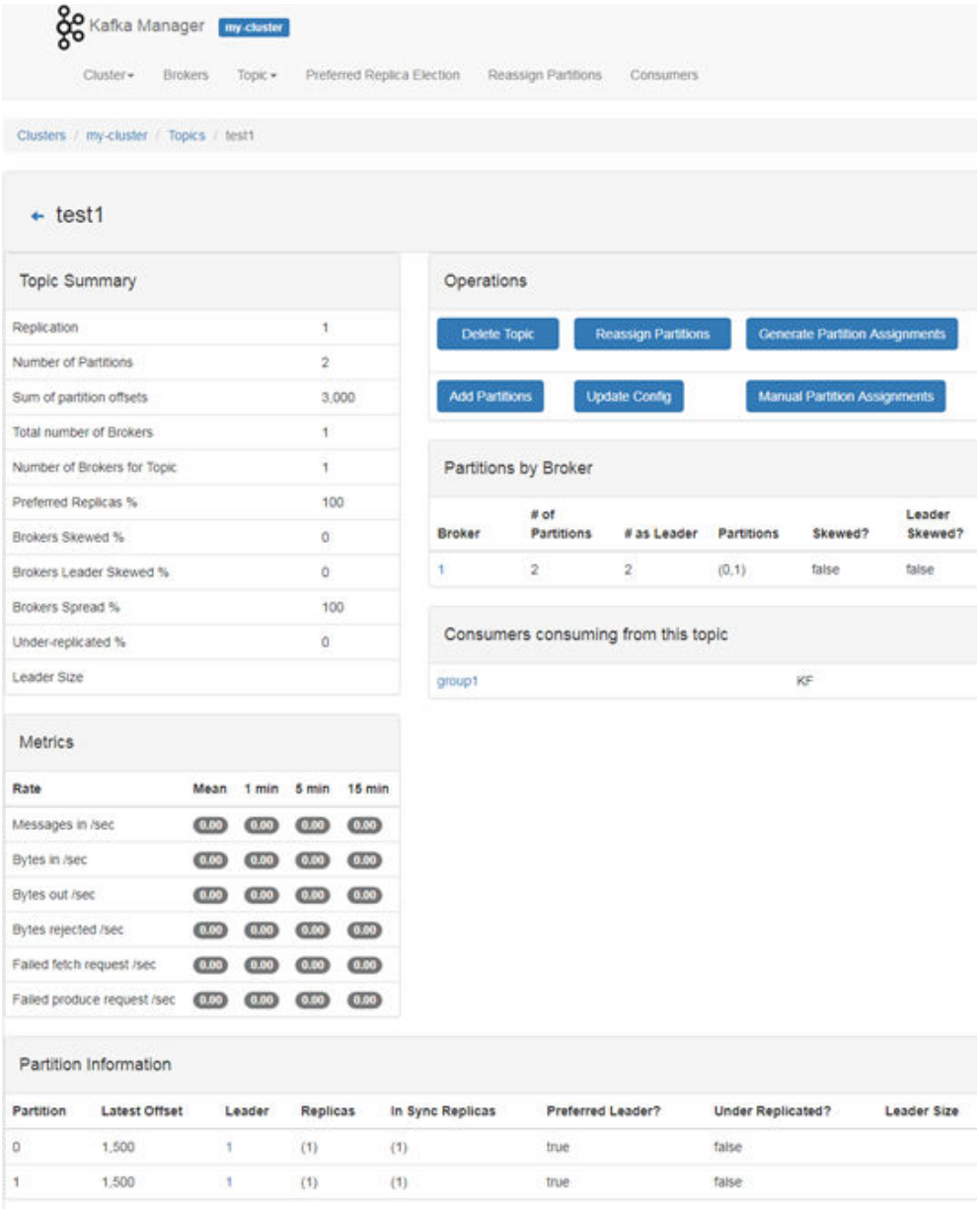
Step 3 Choose **Topic** > **List** to view the topic list of the current cluster and information about each topic.

Figure 12-28 Topic list



Step 4 Click a topic name to view details about the topic.

Figure 12-29 Topic details



----End

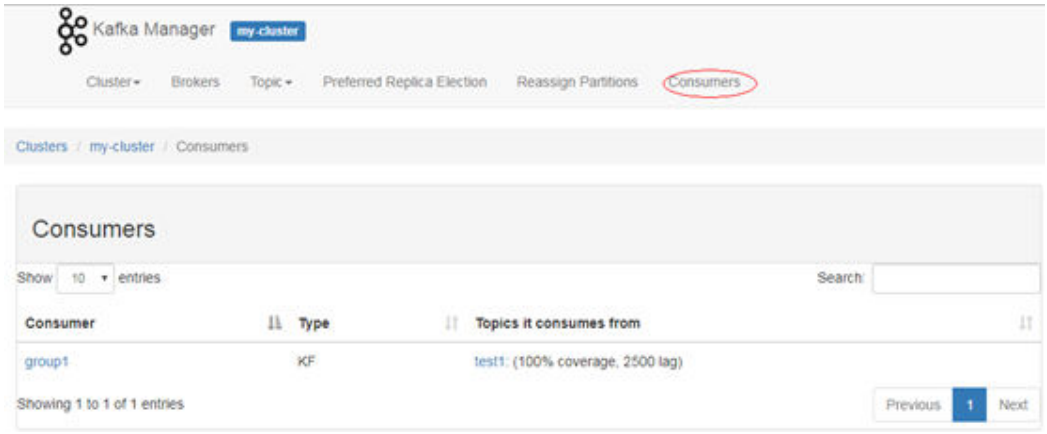
Viewing Consumers Information

Step 1 Log in to the KafkaManager web UI.

Step 2 On the cluster list page, click a cluster name to access the **Summary** page of the cluster.

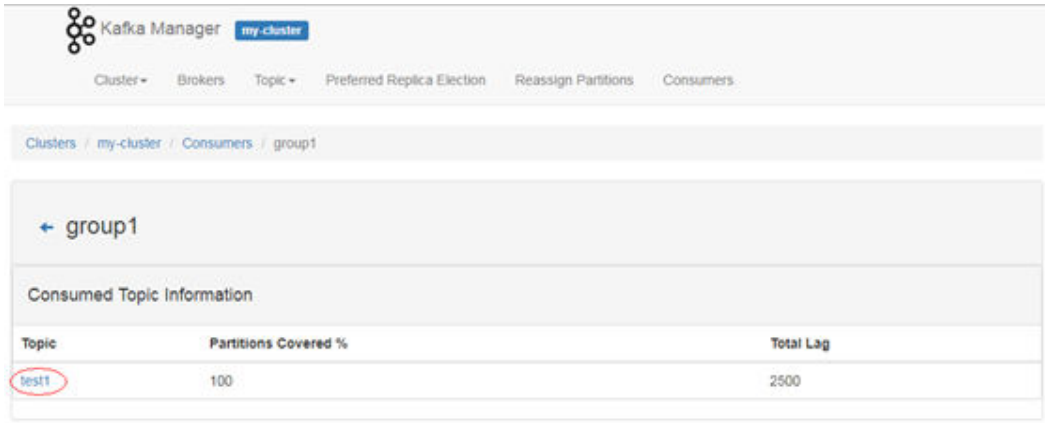
Step 3 Click **Consumers** to view the consumers of the current cluster and each consumer's consumption information.

Figure 12-30 Consumers



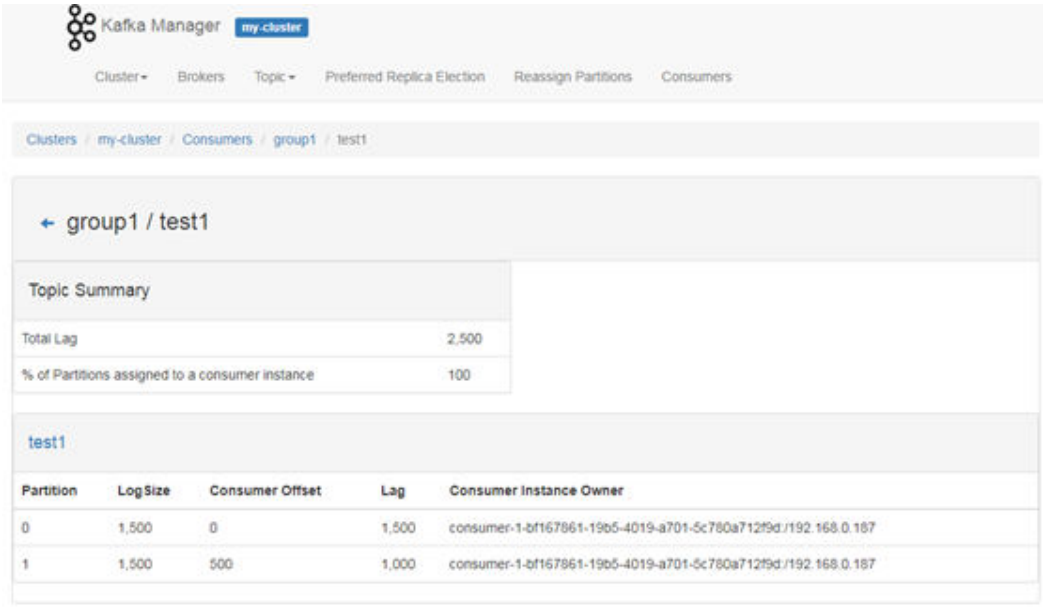
Step 4 Click a consumer name to view the list of the consumed topics.

Figure 12-31 List of topics consumed by the consumer



Step 5 Click a topic name in the topic list of the consumer to view consumption information about the topic.

Figure 12-32 Topic consumption details



----End

Modifying the Partition of a Topic Through KafkaManager

Step 1 Log in to the KafkaManager web UI.

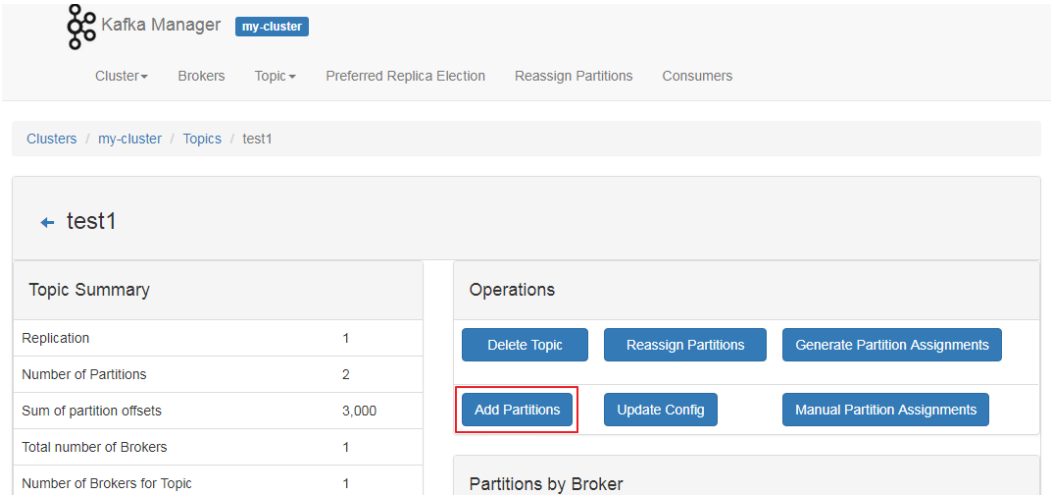
Step 2 On the cluster list page, click a cluster name to access the **Summary** page of the cluster.

Step 3 Choose **Topic** > **List** to access the topic list page of the current cluster.

Step 4 Click a topic name to access the **Topic Summary** page.

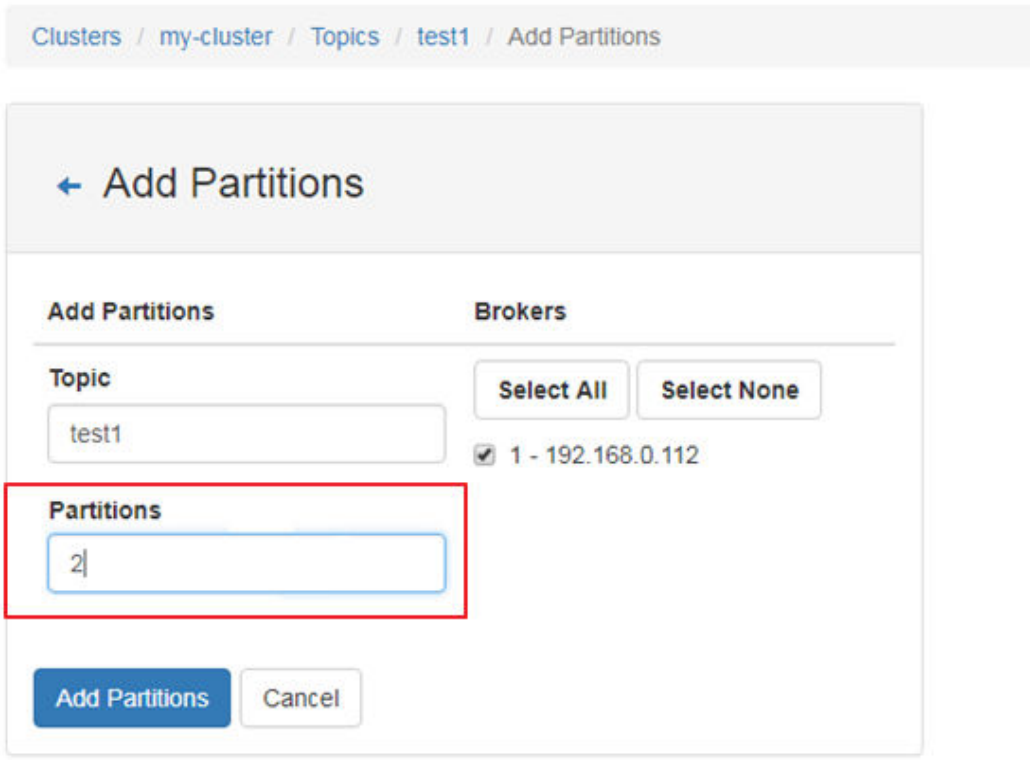
Step 5 Click **Add Partitions**. The page for adding partitions is displayed.

Figure 12-33 Adding partitions



Step 6 Confirm the topic name and modify the value of the **Partitions** parameter and click **Add Partitions** to add partitions.

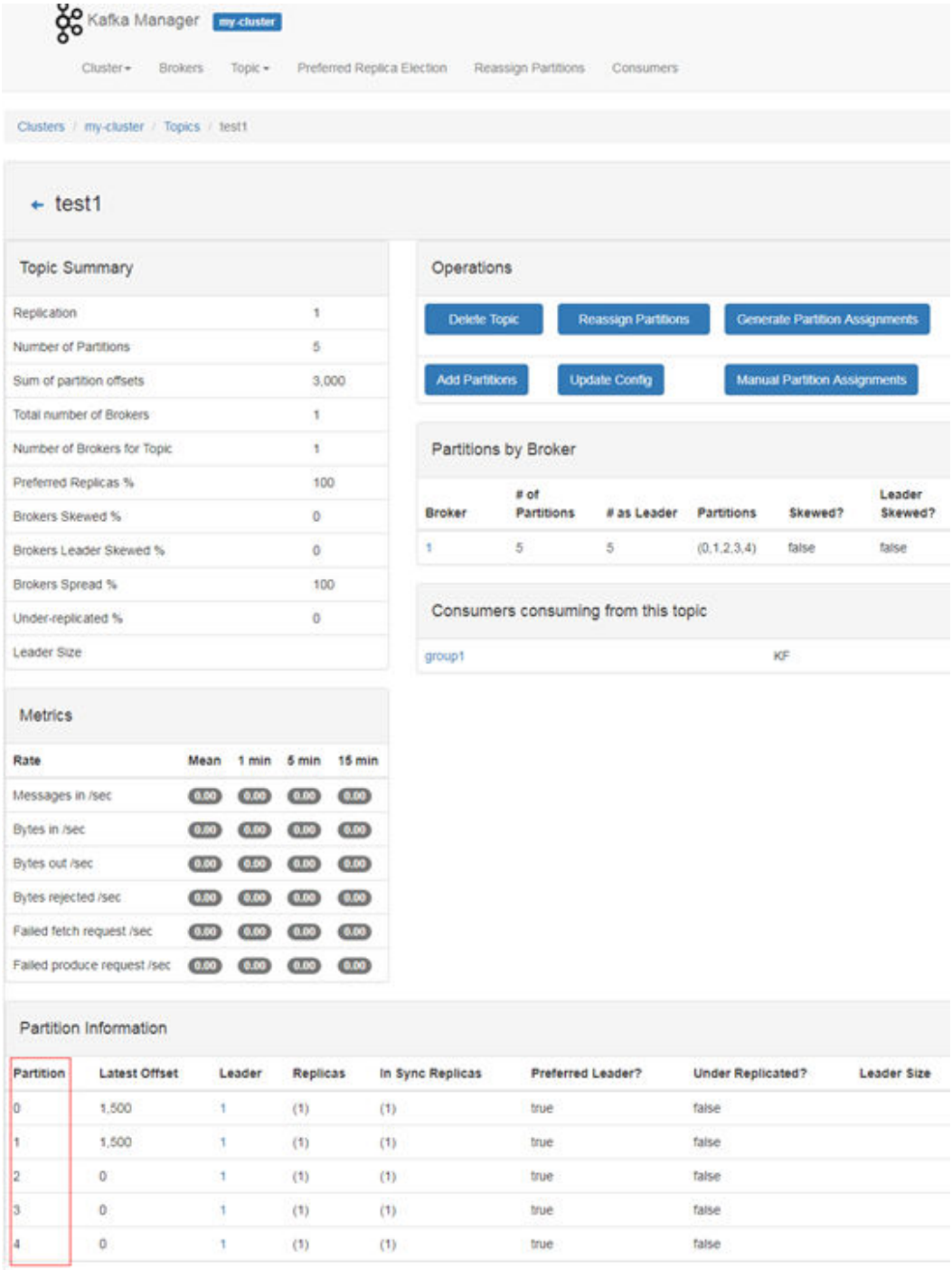
Figure 12-34 Modifying the number of partitions



Step 7 After the partitions are added successfully, click **Go to topic view** to return to the **Topic Summary** page.

Step 8 Check the number of partitions in **Partition Information** in the lower part of the **Topic Summary** page.

Figure 12-35 Partition Information



Step 9 (Optional) If you are not satisfied with the assigned partitions, you can use the partition reassignment function to automatically reassign partitions.

1. On the **Topic Summary** page, click **Generate Partition Assignments**.
2. Select the broker instance and click **Generate Partition Assignments** to generate a partition.
3. After partition generation, click **Go to topic view** to return to the **Topic Summary** page.

4. On the **Topic Summary** page, click **Reassign Partitions** to automatically assign partitions to the broker instance of the cluster.
5. Click **Go to reassign partitions** to view details about the reassigned partitions.

Step 10 (Optional) If you are not satisfied with the automatically assigned partitions, you can manually assign the partitions.

1. On the **Topic Summary** page, click **Manual Partition Assignments** to access the page for manually assign partitions.
2. Manually assign a broker ID to each partition replica, and click **Save Partition Assignment** to save the changes.
3. Click **Go to topic view** to return to the **Topic Summary** page and view the partition details.

----End

12.16 Using Kudu

12.16.1 Using Kudu from Scratch

Kudu is a columnar storage manager developed for the Apache Hadoop platform. Kudu shares the common technical properties of Hadoop ecosystem applications. It is horizontally scalable and supports highly available operations.

Prerequisites

The cluster client has been installed. For example, the client is installed in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it to the actual installation directory.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the Kudu command line tool.

Run the command line tool of the Kudu component to view help information.

kudu -h

The command output is as follows:

```
Usage: kudu <command> [<args>]

<command> can be one of the following:
     cluster   Operate on a Kudu cluster
    diagnose   Diagnostic tools for Kudu servers and clusters
```

```
          fs   Operate on a local Kudu filesystem
         hms   Operate on remote Hive Metastores
local_replica   Operate on local tablet replicas via the local filesystem
      master   Operate on a Kudu Master
         pbc   Operate on PBC (protobuf container) files
        perf   Measure the performance of a Kudu cluster
remote_replica   Operate on remote tablet replicas on a Kudu Tablet Server
       table   Operate on Kudu tables
      tablet   Operate on remote Kudu tablets
        test   Various test actions
     tserver   Operate on a Kudu Tablet Server
         wal   Operate on WAL (write-ahead log) files
```

NOTE

The Kudu command line tool does not support DDL and DML operations, but provides the refined query function for the **cluster**, **master**, **tserver**, **fs**, and **table** parameters.

Common operations:

- Check the tables in the current cluster.

  kudu table list *KuduMaster instance IP1:7051, KuduMaster instance IP2:7051, KuduMaster instance IP3:7051*

- Query the configurations of the KuduMaster instance of the Kudu service.

  kudu master get_flags *KuduMaster instance IP:7051*

- Query the schema of a table.

  kudu table describe *KuduMaster instance IP1:7051, KuduMaster instance IP2:7051, KuduMaster instance IP3:7051 Table name*

- Delete a table.

  kudu table delete *KuduMaster instance IP1:7051, KuduMaster instance IP2:7051, KuduMaster instance IP3:7051 Table name*

  NOTE

  To obtain the IP address of the KuduMaster instance, choose **Components** > **Kudu** > **Instances** on the cluster details page.

----End

12.16.2 Accessing the Kudu Web UI

You can view Kudu job information on the Kudu web UI.

Prerequisites

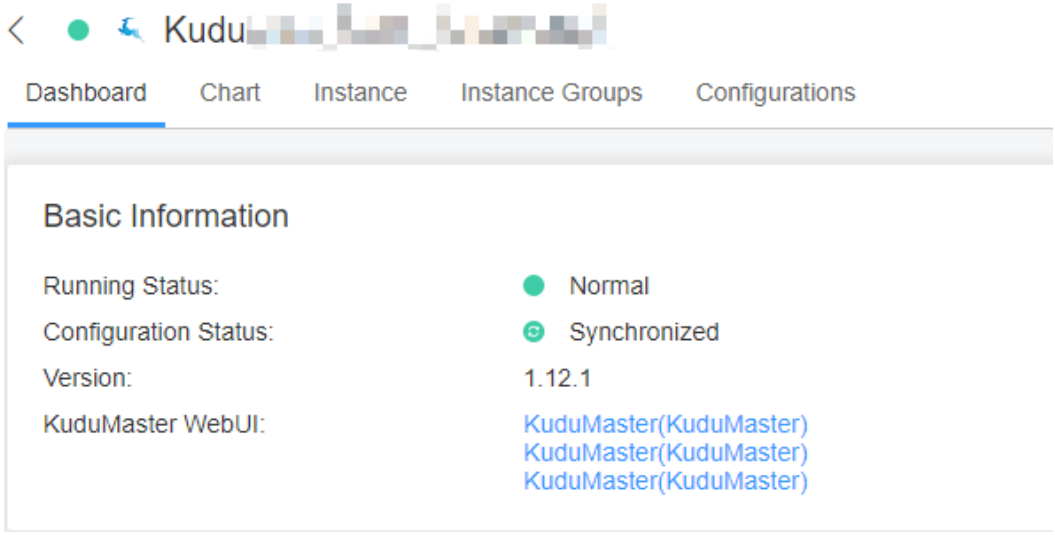
Kudu has been installed in a cluster.

Accessing KuduMaster WebUI (MRS 3.x or Later)

Step 1 Log in to Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 2 Choose **Cluster** > **Services**> **Spark**.

Step 3 In the **Dashboard** page of Kudu, click **KuduMaster(KuduMaster)** on the right side of **KuduMaster WebUI**. The KuduMaster web UI is displayed.



----End

Accessing KuduMaster WebUI (Versions Earlier Than MRS 3.*x*)

Step 1 Access Manager. For details, see **Accessing MRS Manager (Versions Earlier Than MRS 3.x)**.

Step 2 choose **Services** > **Kudu**.

Step 3 In **KuduMaster WebUI** of **Kudu Summary**, click **KuduMaster(KuduMaster)**. The KuduMaster web UI is displayed.

----End

12.17 Using Loader

12.17.1 Using Loader from Scratch

You can use Loader to import data from the SFTP server to HDFS.

This section applies to versions earlier than MRS 3.x.

Prerequisites

- You have prepared service data.
- You have created an analysis cluster.

Procedure

Step 1 Access the Loader page.

1. Go to the cluster details page and choose **Services**.
2. Choose **Hue**. In **Hue Web UI** of **Hue Summary**, click **Hue (Active)**. The Hue web UI is displayed.
3. Choose **Data Browsers** > **Sqoop**.

The job management tab page is displayed by default on the Loader page.

Step 2 On the Loader page, click **Manage links**.

Step 3 Click **New link** and create **sftp-connector**. For details, see **File Server Link**.

Step 4 Click **New link**, enter the link name, select **hdfs-connector**, and create **hdfs-connector**.

Step 5 On the Loader page, click **Manage jobs**.

Step 6 Click **New Job**.

Step 7 In **Connection**, set parameters.

1. In **Name**, enter a job name.
2. Select the source link created in **Step 3** and the target link created in **Step 4**.

Step 8 In **From**, configure the job of the source link.

For details, see **ftp-connector or sftp-connector**.

Step 9 In **To**, configure the job of the target link.

For details, see **hdfs-connector**.

Step 10 In **Task Config**, set job running parameters.

Table 12-274 Loader job running properties

Parameter	Description
Extractors	Number of Map tasks
Loaders	Number of Reduce tasks This parameter is displayed only when the destination field is HBase or Hive.
Max. Error Records in a Single Shard	Error record threshold. If the number of error records of a single Map task exceeds the threshold, the task automatically stops and the obtained data is not returned. **NOTE** Data is read and written in batches for **MYSQL** and **MPPDB** of **generic-jdbc-connector** by default. Errors are recorded once at most for each batch of data.
Dirty Data Directory	Directory for saving dirty data. If you leave this parameter blank, dirty data will not be saved.

Step 11 Click **Save**.

----End

12.17.2 How to Use Loader

This section applies to versions earlier than MRS 3.x.

Process

The process for migrating user data with Loader is as follows:

1. Access the Loader page of the Hue web UI.
2. Manage Loader links.
3. Create a job and select a data source link and a link for saving data.
4. Run the job to complete data migration.

Loader Page

The Loader page is a graphical data migration management tool based on the open source Sqoop web UI and is hosted on the Hue web UI. Perform the following operations to access the Loader page:

1. Access the Hue web UI. For details, see **Accessing the Hue Web UI**.
2. Choose **Data Browsers** > **Sqoop**.

   The job management tab page is displayed by default on the Loader page.

Loader Links

Loader links save data location information. Loader uses links to access data or save data to the specified location. Perform the following operations to access the Loader link management page:

1. Access the Loader page.
2. Click **Manage links**.

   The Loader link management page is displayed.

   Click **Manage jobs** to return to the job management page.
3. Click **New link** to go to the configuration page and set parameters to create a Loader link.

Loader Jobs

Loader jobs are used to manage data migration tasks. Each job consists of a source data link and a destination data link. A job reads data from the source link and saves data to the destination link to complete a data migration task.

12.17.3 Loader Link Configuration

This section applies to versions earlier than MRS 3.x.

Overview

Loader supports the following links. This section describes configurations of each link.

- obs-connector
- generic-jdbc-connector
- ftp-connector or sftp-connector

- hbase-connector, hdfs-connector, or hive-connector

OBS Link

An OBS link is a data exchange channel between Loader and OBS. **Table 12-275** describes the configuration parameters.

Table 12-275 obs-connector configuration

Parameter	Description
Name	Name of a Loader connection
OBS Server	Enter an OBS endpoint. The common format is **OBS.*Region.DomainName***. Run the following command to query the endpoints of OBS: **cat /opt/Bigdata/apache-tomcat-7.0.78/webapps/web/WEB-INF/classes/cloud-obs.properties**
Port	Specifies the port for accessing OBS data. The default value is **443**.
Access Key	AK for a user to access OBS
Security Key	SK corresponding to AK

Relational Database Link

A relational database link is a data exchange channel between Loader and a relational database. **Table 12-276** describes the configuration parameters.

NOTE

Some parameters are hidden by default. They appear only after you click **Show Senior Parameter**.

Table 12-276 generic-jdbc-connector configuration

Parameter	Description
Name	Name of a Loader link
Database Type	Data types supported by Loader links: **ORACLE**, **MYSQL**, and **MPPDB**
Host	Database access address, which can be an IP address or domain name.
Port	Port for accessing the database
Database	Name of the database saving data
Username	Username for accessing the database

Parameter	Description
Password	Password of the user Use the actual password.

Table 12-277 Senior parameter configuration

Parameter	Description
Fetch Size	A maximum volume of data obtained during each database access
Connection Properties	Drive properties exclusive to the database link supported by databases of different types, for example, **autoReconnect** of MYSQL. If you want to define the drive properties, click **Add**.
Identifier Enclose	Delimiter for reserving keywords in the database SQL. Delimiters defined in different databases vary.

File Server Link

File server links include FTP and SFTP links and serve as a data exchange channel between Loader and a file server. **Table 12-278** describes the configuration parameters.

Table 12-278 ftp-connector or **sftp-connector** configuration

Parameter	Description
Name	Name of a Loader link
Hostname/IP	Enter the file server access address, which can be a host name or IP address.
Port	Port for accessing the file server. • Use port **21** for FTP. • Use port **22** for SFTP.
Username	Username for logging in to the file server
Password	Password of the user

MRS Cluster Link

MRS cluster links include HBase, HDFS, and Hive links and serve as a data exchange channel between Loader and HBase, HDFS, or Hive.

When configuring an MRS cluster link, set the name, select a connector, for example, **hbase-connector**, **hdfs-connector**, or **hive-connector**, and save the settings.

12.17.4 Managing Loader Links (Versions Earlier Than MRS 3.x)

Scenario

You can create, view, edit, and delete links on the Loader page.

This section applies to versions earlier than MRS 3.x.

Prerequisites

You have accessed the Loader page. For details, see **Loader Page**.

Creating a Link

Step 1 On the Loader page, click **Manage links**.

Step 2 Click **New link** and configure link parameters.

For details about the parameters, see **Loader Link Configuration**.

Step 3 Click **Save**.

If link configurations, for example, IP address, port, and access user information, are incorrect, the link will fail to be verified and saved. In addition, VPC configurations may affect the network connectivity.

NOTE

You can click **Test** to immediately check whether the link is available.

----End

Viewing a Link

Step 1 On the Loader page, click **Manage links**.

- If Kerberos authentication is enabled for the cluster, all links created by the current user are displayed by default and other users' links cannot be displayed.
- If Kerberos authentication is disabled for the cluster, all Loader links of the cluster are displayed.

Step 2 In **Sqoop Links**, enter a link name to filter the link.

----End

Editing a Link

Step 1 On the Loader page, click **Manage links**.

Step 2 Click the link name to go to the edit page.

Step 3 Modify the link configuration parameters based on service requirements.

Step 4 Click **Test**.

If the test is successful, go to **Step 5**. If a message displays indicating that OBS server cannot be connected, repeat **Step 3**.

Step 5 Click **Save**.

If a Loader job has integrated into a Loader link, editing the link parameters may affect Loader running.

----End

Deleting a Link

Step 1 On the Loader page, click **Manage links**.

Step 2 Locate the row that contains the target link, and click **Delete**.

Step 3 In the dialog box, click **Yes, delete it**.

If a Loader job has integrated a Loader link, the link cannot be deleted.

----End

12.17.5 Source Link Configurations of Loader Jobs

Overview

When Loader jobs obtain data from different data sources, a link corresponding to a data source type needs to be selected and the link properties need to be configured.

This section applies to versions earlier than MRS 3.x.

obs-connector

Table 12-279 Data source link properties of **obs-connector**

Parameter	Description
Bucket Name	OBS file system for storing source data.
Source Directory/ File	Actual storage form of source data. It can be either all data files in a directory or a single data file contained in the file system.
File Format	Loader supports the following file formats of data stored in OBS: • **CSV_FILE**: Specifies a text file. When the destination link is a database link, only the text file is supported. • **BINARY_FILE**: Specifies binary files excluding text files.
Line Separator	Identifier of each line end of source data
Field Separator	Identifier of each field end of source data

Parameter	Description
Encoding Type	Text encoding type of source data. It takes effect on text files only.
File Split Type	The following types are supported: ● **File**: The number of files is assigned to a map task by the total number of files. The calculation formula is **Total number of files/Extractors**. ● **Size**: A file size is assigned to a map task by the total file size. The calculation formula is **Total file size/ Extractors**.

generic-jdbc-connector

Table 12-280 Data source link properties of **generic-jdbc-connector**

Parameter	Description
Schema/ Tablespace	Name of the database storing source data. You can query and select it on the interface.
Table Name	Data table storing the source data. You can query and select it on the interface.
Partition Column	If multiple columns need to be read, use this column to split the result and obtain data.
Where Clause	Query statement used when accessing the database

ftp-connector or sftp-connector

Table 12-281 Data source link properties of **ftp-connector** or **sftp-connector**

Parameter	Description
Source Directory/ File	Actual storage form of source data. It can be either all data files in a directory or single data file contained in the file server.
File Format	Loader supports the following file formats of data stored in the file server: ● **CSV_FILE**: Specifies a text file. When the destination link is a database link, only the text file is supported. ● **BINARY_FILE**: Specifies binary files excluding text files.

Parameter	Description
Line Separator	Identifier of each line end of source data **NOTE** If FTP or SFTP serves as a source link and **File Format** is set to **BINARY_FILE**, the value of **Line Separator** in the advanced properties is invalid.
Field Separator	Identifier of each field end of source data **NOTE** If FTP or SFTP serves as a source link and **File Format** is set to **BINARY_FILE**, the value of **Field Separator** in the advanced properties is invalid.
Encoding Type	Text encoding type of source data. It takes effect on text files only.
File Split Type	The following types are supported: ● **File**: The number of files is assigned to a map task by the total number of files. The calculation formula is **Total number of files/Extractors**. ● **Size**: A file size is assigned to a map task by the total file size. The calculation formula is **Total file size/ Extractors**.

hbase-connector

Table 12-282 Data source link properties of **hbase-connector**

Parameter	Description
Table Name	HBase table storing source data

hdfs-connector

Table 12-283 Data source link properties of **hdfs-connector**

Parameter	Description
Source Directory/ File	Actual storage form of source data. It can be either all data files in a directory or single data file contained in HDFS.
File Format	Loader supports the following file formats of data stored in HDFS: ● **CSV_FILE**: Specifies a text file. When the destination link is a database link, only the text file is supported. ● **BINARY_FILE**: Specifies binary files excluding text files.

Parameter	Description
Line Separator	Identifier of each line end of source data **NOTE** If HDFS serves as a source link and **File Format** is set to **BINARY_FILE**, the value of **Line Separator** in the advanced properties is invalid.
Field Separator	Identifier of each field end of source data **NOTE** If HDFS serves as a source link and **File Format** is set to **BINARY_FILE**, the value of **Field Separator** in the advanced properties is invalid.
File Split Type	The following types are supported: • **File**: The number of files is assigned to a map task by the total number of files. The calculation formula is **Total number of files/Extractors**. • **Size**: A file size is assigned to a map task by the total file size. The calculation formula is **Total file size/ Extractors**.

hive-connector

Table 12-284 Data source link properties of **hive-connector**

Parameter	Description
Database Name	Name of the Hive database storing the data source. You can query and select it on the interface.
Table	Name of the Hive table storing the data source. You can query and select it on the interface.

12.17.6 Destination Link Configurations of Loader Jobs

Overview

When Loader jobs save data to different storage locations, a destination link needs to be selected and the link properties need to be configured.

obs-connector

Table 12-285 Destination link properties of **obs-connector**

Parameter	Description
Bucket Name	OBS file system for storing final data.

Parameter	Description
Output Directory	Directory for storing final data in the file system. A directory must be specified.
File Format	Loader supports the following file formats of data stored in OBS: • **CSV_FILE**: Specifies a text file. When the destination link is a database link, only the text file is supported. • **BINARY_FILE**: Specifies binary files excluding text files.
Line Separator	Identifier of each line end of final data
Field Separator	Identifier of each field end of final data
Encoding Type	Text encoding type of final data. It takes effect on text files only.

generic-jdbc-connector

Table 12-286 Destination link properties of **generic-jdbc-connector**

Parameter	Description
Schema Name	Name of the database storing final data
Table	Name of the table saving final data

ftp-connector or sftp-connector

Table 12-287 Destination link properties of **ftp-connector** or **sftp-connector**

Parameter	Description
Output Directory	Directory for storing final data in the file server. A directory must be specified.
File Format	Loader supports the following file formats of data stored in the file server: • **CSV_FILE**: Specifies a text file. When the destination link is a database link, only the text file is supported. • **BINARY_FILE**: Specifies binary files excluding text files.
Line Separator	Identifier of each line end of final data **NOTE** If FTP or SFTP serves as a destination link and **File Format** is set to **BINARY_FILE**, the value of **Line Separator** in the advanced properties is invalid.

Parameter	Description
Field Separator	Identifier of each field end of final data **NOTE** If FTP or SFTP serves as a destination link and **File Format** is set to **BINARY_FILE**, the value of **Field Separator** in the advanced properties is invalid.
Encoding Type	Text encoding type of final data. It takes effect on text files only.

hbase-connector

Table 12-288 Destination link properties of **hbase-connector**

Parameter	Description
Table Name	Name of the HBase table saving final data. You can query and select it on the interface.
Method	Data can be imported to an HBase table using either **BULKLOAD** or **PUTLIST**.
Clear Data Before Import	Whether to clear data in the destination HBase table. Options are as follows: ● **True**: Clean up data in the table. ● **False**: Do not clean up data in the table. If you select **False**, an error is reported during job running if data exists in the table.

hdfs-connector

Table 12-289 Destination link properties of **hdfs-connector**

Parameter	Description
Output Directory	Directory for storing final data in HDFS. A directory must be specified.
File Format	Loader supports the following file formats of data stored in HDFS: ● **CSV_FILE**: Specifies a text file. When the destination link is a database link, only the text file is supported. ● **BINARY_FILE**: Specifies binary files excluding text files.
Compression Codec	Compression mode used when a file is saved to HDFS. The following modes are supported: **NONE**, **DEFLATE**, **GZIP**, **BZIP2**, **LZ4**, and **SNAPPY**.

Parameter	Description
Overwrite	How to process files in the output directory when files are imported to HDFS. Options are as follows: • **True**: Clean up files in the directory and import new files by default. • **False**: Do not clean up files. If files exist in the output directory, job running fails.
Line Separator	Identifier of each line end of final data **NOTE** If HDFS serves as a destination link and **File Format** is set to **BINARY_FILE**, the value of **Line Separator** in the advanced properties is invalid.
Field Separator	Identifier of each field end of final data **NOTE** If HDFS serves as a destination link and **File Format** is set to **BINARY_FILE**, the value of **Field Separator** in the advanced properties is invalid.

hive-connector

Table 12-290 Destination link properties of **hive-connector**

Parameter	Description
Database	Name of the Hive database storing final data. You can query and select it on the interface.
Table	Name of the Hive table saving final data. You can query and select it on the interface.

12.17.7 Managing Loader Jobs

Scenario

You can create, view, edit, and delete jobs on the Loader page.

This section applies to versions earlier than MRS 3.x.

Prerequisites

You have accessed the Loader page. For details, see **Loader Page**.

Creating a Job

Step 1 On the Loader page, click **New job**.

Step 2 In **Connection**, set parameters.

1. In **Name**, enter a job name.
2. In **From link** and **To link**, select links accordingly.

   After you select a link of a type, data is obtained from the specified source and saved to the destination.

   NOTE

   If no available link exists, click **Add a new link**.

Step 3 In **From**, configure the job of the source link.

For details, see **Source Link Configurations of Loader Jobs**.

Step 4 In **To**, configure the job of the destination link.

For details, see **Destination Link Configurations of Loader Jobs**.

Step 5 Check whether a database link is selected in **To link**.

Database links include:

- generic-jdbc-connector
- hbase-connector
- hive-connector

If you set **To link** to a database link, you need to configure a mapping between service data and a field in the database table.

- If you set it to a database link, go to **Step 6**.
- If you do not set it to a database link, go to **Step 7**.

Step 6 In **Field Mapping**, enter a field mapping. Then proceed to **Step 7**.

Field Mapping specifies a mapping between each column of user data and a field in the database table.

Table 12-291 Field Mapping properties

Parameter	Description
Column Num	Field sequence of service data
Sample	First row of sample values of service data
Column Family	When **To link** is **hbase-connector**, you can select a column family for storing data.
Destination Field	Field for storing data
Type	Type of the field selected by the user
Row Key	When **To link** is **hbase-connector**, you need to select **Destination Field** as a row key.

NOTE

If the value of **From** is a connector of a file type, for example, SFTP, FTP, OBS, and HDFS files, the value of **Field Mapping** is the first row of data in the file. Ensure that the first row of data is complete. Otherwise, the Loader job will not extract columns that are not mapped.

Step 7 In **Task Config**, set job running parameters.

Table 12-292 Loader job running properties

Parameter	Description
Extractors	Number of Map tasks
Loaders	Number of Reduce tasks This parameter is displayed only when the destination field is HBase or Hive.
Max. Error Records in a Single Shard	Error record threshold. If the number of error records of a single Map task exceeds the threshold, the task automatically stops and the obtained data is not returned. **NOTE** Data is read and written in batches for **MYSQL** and **MPPDB** of **generic-jdbc-connector** by default. Errors are recorded once at most for each batch of data.
Dirty Data Directory	Specifies the directory for saving dirty data. If you leave this parameter blank, dirty data will not be saved.

Step 8 Click **Save**.

----End

Viewing a Job

Step 1 Access the Loader page. The Loader job management page is displayed by default.

- If Kerberos authentication is enabled for the cluster, all jobs created by the current user are displayed by default and other users' jobs cannot be displayed.
- If Kerberos authentication is disabled for the cluster, all Loader jobs of the cluster are displayed.

Step 2 In **Sqoop Jobs**, enter a job name to filter the job.

Step 3 Click **Refresh** to obtain the latest job status.

----End

Editing a Job

Step 1 Access the Loader page. The Loader job management page is displayed by default.

Step 2 Click the job name to go to the edit page.

Step 3 Modify the job configuration parameters based on service requirements.

Step 4 Click **Save**.

NOTE

Basic job operations in the navigation bar on the left are **Run**, **Copy**, **Delete**, **Disable**, **History Record**, and **Show Job JSON Definition**.

----End

Deleting a Job

Step 1 Access the Loader page.

Step 2 In the row of the specified job, click ✖.

You can also select one or more jobs and click **Delete** Job in the upper right corner of the job list.

Step 3 In the dialog box, click **Yes, delete it**.

If the state of a Loader job is **Running**, the job fails to be deleted.

----End

12.17.8 Preparing a Driver for MySQL Database Link

Scenario

As a component for batch data export, Loader can import and export data using a relational database.

Prerequisites

You have prepared service data.

Procedure

Procedure for versions earlier than MRS cluster 3.*x*:

Step 1 Download the MySQL JDBC driver **mysql-connector-java-5.1.21.jar** from the MySQL official website. For details about how to select the MySQL JDBC driver, see the following table.

Table 12-293 Version information

JDBC Driver Version	MySQL Version
Connector/J 5.1	MySQL 4.1, MySQL 5.0, MySQL 5.1, and MySQL 6.0 alpha
Connector/J 5.0	MySQL 4.1, MySQL 5.0 servers, and distributed transaction (XA)

JDBC Driver Version	MySQL Version
Connector/J 3.1	MySQL 4.1, MySQL 5.0 servers, and MySQL 5.0 except distributed transaction (XA)
Connector/J 3.0	MySQL 3.x and MySQL 4.1

Step 2 Upload **mysql-connector-java-5.1.21.jar** to the Loader installation directory on the active and standby MRS Master nodes.

- For versions earlier than MRS 3.x, upload the package to **/opt/Bigdata/MRS_XXX/install/FusionInsight-Sqoop-1.99.7/FusionInsight-Sqoop-1.99.7/server/jdbc/**.

  In the preceding path, **XXX** indicates the MRS version number. Change it based on site requirements.

Step 3 Change the owner of the **mysql-connector-java-5.1.21.jar** package to **omm:wheel**.

Step 4 Modify the **jdbc.properties** configuration file.

Change the key value of **MYSQL** to **mysql-connector-java-5.1.21.jar**, for example, **MYSQL=mysql-connector-java-5.1.21.jar**.

Step 5 Restart the Loader service.

----End

Procedure for MRS cluster 3.*x* and later versions:

Modify the permission on the JAR package of the relational database driver.

Step 1 Log in to the active and standby management nodes of the Loader service, obtain the driver JAR package of the relational database, and save it to the following directory on the active and standby Loader nodes: **${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Sqoop-1.99.3/FusionInsight-Sqoop-1.99.3/server/webapps/loader/WEB-INF/ext-lib**

NOTE

The version 8.1.0.1 is used as an example. Replace it with the actual version number.

Step 2 Run the following commands as user **root** on the active and standby nodes of the Loader service to change the permission:

cd ${BIGDATA_HOME}/FusionInsight_Porter_8.1.0.1/install/FusionInsight-Sqoop-1.99.3/FusionInsight-Sqoop-1.99.3/server/webapps/loader/WEB-INF/ext-lib

chown omm:wheel *JAR package name*

chmod 600 *JARpackage name*

Step 3 Log in to FusionInsight Manager. Choose **Cluster** and click the target cluster name. In the navigation pane on the left, choose **Services** > **Loader**. In the upper

right corner, choose **More**, select **Restart Service**, and enter the password of the administrator to restart the Loader service.

----End

12.17.9 Loader Log Overview

Log Description

Log path: The default storage path of Loader log files is **/var/log/Bigdata/loader/Log category**.

- runlog: /var/log/Bigdata/loader/runlog (run logs)
- scriptlog: /var/log/Bigdata/loader/scriptlog/ (script execution logs)
- catalina: /var/log/Bigdata/loader/catalina (Tomcat startup and stop logs)
- audit: /var/log/Bigdata/loader/audit (audit logs)

Log archive rule:

The automatic compression and archiving function are enabled for Loader run logs and audit logs. By default, when the size of a log file exceeds 10 MB, the log file is automatically compressed into a log file named in the following rule: *<Original log file name>-<yyyy-mm-dd_hh-mm-ss>.[ID]***.log.zip**. A maximum of 20 latest compressed files are reserved. The number of compressed files can be configured on the Manager portal.

Table 12-294 Loader log list

Log Type	Log File Name	Description
Run log	loader.log	Loader system log file that records most of the logs generated when the TelcoFS system is running.
	loader-omm-***-pid***-gc.log.*.current	Loader process GC log file
	sqoopInstanceCheck.log	Loader instance health check log file
Audit log	default.audit	Loader operation audit log file that records operations such as adding, deleting, modifying, and querying jobs and user login
Tomcat log	catalina.out	Tomcat run log file.
	catalina. <yyyy-mm-dd >.log	Tomcat run log file
	host-manager. <yyyy-mm-dd >.log	Tomcat run log file

Log Type	Log File Name	Description
	localhost_access_log. <yyyy-mm-dd >.txt	Tomcat run log file
	manager <yyyy-mm-dd >.log	Tomcat run log file
	localhost. <yyyy-mm-dd >.log	Tomcat run log file
Script log	postInstall.log	Loader installation script log file Log file generated during the execution of the Loader installation script (**postInstall.sh**)
	preStart.log	Pre-startup script log file of the Loader service During startup of the Loader service, a series of preparation operations are first performed (by executing **preStart.sh**), such as generating the keytab file. This log file records information about these operations
	loader_ctl.log	Log file generated when Loader executes the service start and stop script (**sqoop.sh**)

Log Level

Table 12-295 describes the log levels provided by Loader. The priorities of log levels are ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-295 Log levels

Level	Description
ERROR	Error information about the current event processing.
WARN	Exception information about the current event processing.

Level	Description
INFO	Normal running status information about the system and events.
DEBUG	System information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of Loader by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

NOTE

The configurations take effect immediately without the need to restart the service.

----End

Log Formats

The following table lists the Loader log formats.

Table 12-296 Log formats

Log Type	Format	Example
Run log	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<Thread that generates the log>|<Message in the log>|<Location of the log event>	2015-06-29 14:54:35,553 | INFO | [localhost-startStop-1] | ConnectionRequestHandler initialized | org.apache.sqoop.handler.ConnectionRequestHandler.<init>(ConnectionRequestHandler.java:100)

Log Type	Format	Example
Audit log	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|default|<Message in the log>|<Location of the log event>	2015-06-29 15:35:40,969 INFO default: UserName=admin, UserIP=10.52.0.111, Time=2015-06-29 15:35:40,969, Operation=submit, Resource=submission@21, Result=Failure, Detail={[reason:GET_SFTP_SESSION_FAILED:Failed to get sftp session - 10.162.0.35 (caused by: Auth cancel)]; [config:null]}

12.17.10 Example: Using Loader to Import Data from OBS to HDFS

Scenario

If you need to import a large volume of data from the external cluster to the internal cluster, import it from OBS to HDFS.

Prerequisites

- You have prepared service data.
- You have created an analysis cluster.

Procedure

Step 1 Upload service data to your OBS file system.

Step 2 Obtain the AK/SK information and create an OBS and HDFS link.

For details, see **Loader Link Configuration**.

Step 3 Access the Loader page.

If Kerberos authentication is enabled in the analysis cluster, refer to instructions in **Accessing the Hue Web UI**.

Step 4 Click **New Job**.

Step 5 In **Information**, set parameters.

1. In **Name**, enter a job name. For example, **obs2hdfs**.
2. In **From link**, select the OBS link you create.
3. In **To link**, select the HDFS link you create.

Step 6 In **From**, set source link parameters.

1. In **Bucket Name**, enter a name of the OBS file system.
2. In **Input directory or file**, enter a detailed location of service data in the file system.

   If it is a single file, enter a complete path containing the file name. If it is a directory, enter the complete path of the directory.
3. In **File format**, enter the type of the service data file.

For details, see **obs-connector**.

Step 7 In **To**, set destination link parameters.

1. In **Output directory**, enter the directory for storing service data in HDFS.

   If Kerberos authentication is enabled in the cluster, the current user accessing Loader needs to have the permission to write data to the directory.
2. In **File format**, enter the type of the service data file.

   The type must correspond to the type in **Step 6.3**.
3. In **Compression codec**, enter a compression algorithm. For example, if you do not compress data, select **NONE**.
4. In **Overwrite**, select **True**.
5. Click **Show Senior Parameter** and set **Line Separator**.
6. Set **Field Separator**.

For details, see **hdfs-connector**.

Step 8 In **Task Config**, set job running parameters.

1. In **Extractors**, enter the number of Map tasks.
2. In **Loaders**, enter the number of Reduce tasks.

   If the destination link is an HDFS link, **Loaders** is hidden.
3. In **Max error records in single split**, enter an error record threshold.
4. In **Dirty data directory**, enter a directory for saving dirty data, for example, **/user/sqoop/obs2hdfs-dd**.

Step 9 Click **Save and execute**.

On the **Manage jobs** page, view the job running result. You can click **Refresh** to obtain the latest job status.

----End

12.17.11 Common Issues About Loader

12.17.11.1 How to Resolve the Problem that Failed to Save Data When Using Internet Explorer 10 or Internet Explorer 11 ?

Question

Internet Explorer 11 or Internet Explorer 10 is used to access the web UI of Loader. After data is submitted, an error occurs.

Answer

- Symptom

  a. When the submitted data is saved, a similar error occurs: Invalid query parameter jobgroup id. cause: [jobgroup].

- Causse

  Some Internet Explorer 11 versions convert POST requests into GET requests after receiving the HTTP 307 response. As a result, POST data cannot be delivered to the server.

- Solution

  Use Google Chrome.

12.17.11.2 Differences Among Connectors Used During the Process of Importing Data from the Oracle Database to HDFS

Question

Three types of connectors are available for importing data from the Oracle database to HDFS using Loader. That is, generic-jdbc-connector, oracle-connector, and oracle-partition-connector. Which one should I select? What are the differences between them?

Answers

- generic-jdbc-connector

  Reads data from the Oracle database in JDBC mode. It is applicable to databases that support JDBC.

  In this mode, data loading performance of Loader is subject to data distribution in a partition column. When data skew occurs (data has only one value or several values) in a partition column, a few Maps process a significant portion of data. As a result, the index becomes invalid, causing a sharp decline in SQL query performance.

  generic-jdbc-connector supports view import and export, but oracle-partition-connector and oracle-connector do not support. Therefore, only this connector can be used to import views.

- Both **oracle-partition-connector** and **oracle-connector**

  can use the ROWID of Oracle for partitioning. oracle-partition-connector is self-developed and oracle-connector is an open-source edition. The two types of connectors share similar performance.

  oracle-connector requires more system table permissions. The following lists the read permissions required by the system tables of **oracle-connector** and **oracle-connector**.

  – **oracle-connector**: dba_tab_partitions, dba_constraints, dba_tables t, dba_segments, v$version, dba_objects, v$instance, SYS_CONTEXT function, dba_extents, and dba_tab_subpartitions

  – **oracle-partition-connector**: DBA_OBJECTS and DBA_EXTENTS

  Compared with **generic-jdbc-connector**, **oracle-partition-connector** and **oracle-connector** have the following advantages:

a. Load balancing: Number and scope of data segments are determined by the storage structure (data blocks) of the source table rather than the data on the source table. In terms of granularity, a data block can occupy a partition.
b. Stable performance: Invalid index faults caused by data skew and bound variable snooping can be completely eliminated.
c. Fast query speed: Using data segmentation delivers a higher query speed than that of using index.
d. Excellent horizontal scalability: The number of generated segments increases with the increase of data volume. In this case, ideal performance can be delivered when you increase the number of concurrent tasks. Contrarily, decreasing concurrent tasks saves resources.
e. Simplified data segmentation logic: Problems like precision loss, type compatibility, and bound variables can be prevented.
f. Enhanced usability: Users do not need to create partition columns and tables for Loader.

12.18 Using MapReduce

12.18.1 Configuring the Log Archiving and Clearing Mechanism

Scenario

Job and task logs are generated during execution of a MapReduce application.

- Job logs are generated by the MRApplicationMaster, which record details about the start and running time of jobs and each task, Counter value, and other information. After being analyzed by HistoryServer, the job logs are used to view job execution details.
- A task log records the log information generated by each task running in a container. By default, task logs are stored only on the local disk of each NodeManager. After the log aggregation function is enabled, the NodeManager merges local task logs and writes them into HDFS after job execution completes.

The job logs and task logs of the MapReduce are stored on HDFS (when the log aggregation function is enabled). If the mechanism for periodically archiving and deleting log files is not configured for a cluster with a large number of computation tasks, the log files will occupy large memory space of HDFS and increase the cluster load.

Log archive is implemented by Hadoop Archives. The number (number of Map tasks) of concurrent archiving tasks started by the Hadoop Archives is related to the total size of log files to be archived. The formula is as follows: Number of concurrent archive tasks = Total size of log files to be archived/Size of archive files.

Configuration

Go to the **All Configurations** page of the MapReduce service. For details, see **Modifying Cluster Service Configuration Parameters**.

Enter a parameter name in the search box. In addition, you need to configure the following information in the **mapred-site.xml** configuration file in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/ directory** on the MapReduce client node:

Table 12-297 Parameter description

Parameter	Description	Default Value
mapreduce.jobhistory.cleaner.enable	Whether to enable the job log file deletion function.	true
mapreduce.jobhistory.cleaner.interval-ms	Period for starting a log file cleanup. Only log files whose retention period is longer than the time specified by **mapreduce.jobhistory.max-age-ms** can be deleted.	86,400,000 ms (1 day)
mapreduce.jobhistory.max-age-ms	Log files whose retention period is longer than the retention period in milliseconds specified by this parameter will be deleted.	1,296,000,000 ms (15 days)

You can configure the following parameters in the **yarn-site.xml** file on the ResourceManager, NodeManager, and MapReduce HistoryServer nodes. The **yarn.nodemanager.remote-app-log-dir** and **yarn.nodemanager.remote-app-log-archive-dir** parameters need to be configured on the Yarn client, and the configurations of the ResourceManager, NodeManager, and MapReduce HistoryServer nodes must be the same as those on the Yarn client.

Table 12-298 Parameter description

Parameter	Description	Default Value
yarn.nodemanager.remote-app-log-dir	Indicates the HDFS path for aggregating the MapReduce job logs.	/tmp/logs
yarn.nodemanager.remote-app-log-archive-dir	Indicates the HDFS path for archiving the MapReduce job logs.	/tmp/archived

Parameter	Description	Default Value
yarn.log-aggregation.archive.files.minimum	Indicates the minimum number of archived MapReduce job log files. The archiving task starts when the number of files in the **yarn.nodemanager.remote-app-log-dir** folder is greater than or equal to the value of this parameter. This parameter applies to MRS 3.*x*.	5,000
yarn.log-aggregation.archive-check-interval-seconds	Indicates the MapReduce job log archiving interval, in seconds. Log files are archived only when the number of log files reaches the value of **yarn.log-aggregation.archive.files.minimum**. The archiving function is disabled when the period is set to **0** or **-1**. This parameter applies to MRS 3.*x*.	-1
yarn.log-aggregation.retain-seconds	Indicates the retention period on HDFS for archiving the MapReduce job logs. The value **-1** indicates that log files are stored permanently.	1,296,000
yarn.log-aggregation.retain-check-interval-seconds	Indicates the check period (in seconds) of the MapReduce job log deletion task. If this parameter is set to **-1**, the check period is one tenth of the log retention period.	86400

12.18.2 Reducing Client Application Failure Rate

Scenario

When the network is unstable or the cluster I/O and CPU are overloaded, client applications might encounter running failures.

Configuration

Adjust the following parameters in the **mapred-site.xml** configuration file on the client to reduce the client application failure rate:

NOTE

The **mapred-site.xml** configuration file is in the **conf** directory of the client installation path, for example, **/opt/client/Yarn/config**.

Table 12-299 Parameter description

Parameter	Description	Default Value
mapreduce.reduce.shuffle.max-host-failures	Indicates the number of allowed failures of an MR task to read remote shuffle data in the Reduce process. When the number is set to be over 5, the client application failure rate can be reduced. This parameter applies to MRS 3.*x*.	5
mapreduce.client.submit.file.replication	Indicates the backup of job files on HDFS. MR tasks are dependent on the job files during running. When the number of backups is set to be over 10, the client application failure rate can be reduced.	10

12.18.3 Transmitting MapReduce Tasks from Windows to Linux

Scenarios

If you want to transmit a job from Windows to Linux, set **mapreduce.app-submission.cross-platform** to **true**. If this parameter is unavailable for a cluster or its value is **false**, the function of transmitting MapReduce tasks from Windows to Linux is not supported. In this case, perform the following operations to add this parameter or change its value to enable this function:

NOTE

This section applies to MRS 3.*x* or later.

Configuration Description

Adjust the following parameter in the **mapred-site.xml** configuration file on the client to enable the running of MapReduce tasks: The **mapred-site.xml** configuration file is in the **config** directory of the client installation path, for example, **/opt/client/Yarn/config**.

Table 12-300 Parameters

Parameter	Description	Default Value
mapreduce.app-submission.cross-platform	Indicates whether to support running of MapReduce tasks after they are transmitted from Windows to Linux. When the parameter value is **true**, the running of MapReduce tasks is supported. When the parameter value is **false**, the running of MapReduce tasks is not supported.	true

12.18.4 Configuring the Distributed Cache

Scenarios

NOTE

This section applies to MRS 3.*x* or later.

Distributed caching is useful in the following scenarios:

Rolling Upgrade

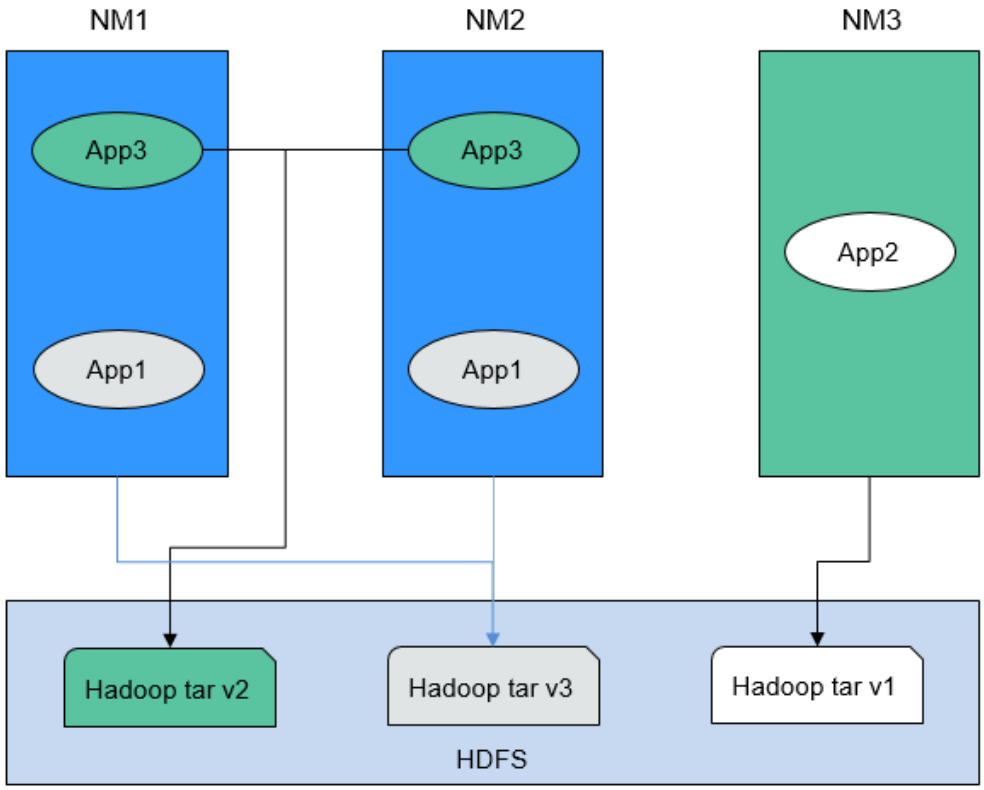
During the upgrade, applications must keep the text content (JAR file or configuration file) unchanged. The content is not based on Yarn of the current version, but on the version when it is submitted. This is a challenging issue. Generally, applications (such as MapReduce, Hive, and Tez) need to be installed locally. Libraries need to be installed on all cluster servers (clients and servers). When a rolling upgrade or downgrade starts in the cluster, the version of the locally installed library changes during application running. During the rolling upgrade, only a few NodeManagers are upgraded first. These NodeManagers obtain the software of the latest version. This leads to inconsistent behavior and can result in run-time errors.

Co-existence of Multiple Yarn Versions

Cluster administrators may run tasks that use multiple versions of Yarn and Hadoop JARs in a cluster. However, this task is difficult to be implemented because the JARs have been localized and have only one version.

The MapReduce application framework can be deployed through the distributed cache and does not depend on the static version copied during installation. Therefore, you can store multiple versions of Hadoop in HDFS and configure the **mapred-site.xml** file to specify the default version used by the task. You can run different versions of MapReduce by setting proper configuration attributes without using the versions deployed in the cluster.

Figure 12-36 Clusters with NodeManagers and Applications of multiple versions



As shown in **Figure 12-36**, the application can use Hadoop JARs in HDFS instead of the local version. Therefore, during the rolling upgrade, even if NodeManager has been upgraded, the application can still run Hadoop of the earlier version.

Configuration Description

Step 1 Save the MapReduce **.tar** package of the specified version to a directory that can be accessed by applications in HDFS, as shown in the following command.

$HADOOP_HOME/**bin/hdfs dfs -put** *hadoop-x.tar.gz /mapred/framework/*

Step 2 Set parameters in the **mapred-site.xml** file based on **Table 12-301**.

Table 12-301 Distributed cache parameters

Parameter	Description	Default Value
mapreduce.application.framework.path	Indicates the URL directing to the archive location. **NOTE** This property can also create an alias for the archive if the URL fragment identity name is specified as follows. In this example, the alias is set to **mr-framework**. <property> <name>mapreduce.application.framework.path</name> <value>hdfs:/mapred/framework/hadoop-x.tar.gz#mr-framework</value> </property>	NA

Parameter	Description	Default Value
mapreduce.application.classpath	Indicates the parameter property, which contains the MapReduce JARs in the class directory. **NOTE** For example, the alias **mr-framework** used in the framework path is used to match the directory. <property> <name>mapreduce.application.classpath</name> <value>$PWD/mr-framework/hadoop/share/hadoop/mapreduce/*:$PWD/mr-framework/hadoop/share/hadoop/mapreduce/lib/*:$PWD/mr-framework/hadoop/share/hadoop/common/*:$PWD/mr-framework/hadoop/share/hadoop/common/lib/*:$PWD/mr-framework/hadoop/share/hadoop/yarn/*:$PWD/mr-framework/hadoop/share/hadoop/yarn/lib/*:$PWD/mr-framework/hadoop/share/hadoop/hdfs/*:$PWD/mr-framework/hadoop/share/hadoop/hdfs/lib/*:/etc/hadoop/conf/secure</value></property>	N/A

You can upload MapReduce tarballs of multiple versions to HDFS. Different **mapred-site.xml** files indicate different locations. After that, you can run tasks for a specific **mapred-site.xml** file. The following is an example of running an MapReduce task for the MapReduce tarball of the *x* version:

hadoop jar *share/hadoop/mapreduce/hadoop-mapreduce-examples-*.jar pi* **-conf etc/hadoop-x/mapred-site.xml** *10 10*

----End

12.18.5 Configuring the MapReduce Shuffle Address

Scenario

When the MapReduce shuffle service is started, it attempts to bind an IP address based on local host. If the MapReduce shuffle service is required to connect to a specific IP address, no configuration is available. The following description allows you to configure a connection to a specific IP address.

Configuration

To bind a specific IP address to the MapReduce shuffle service, set the following parameters in the **mapred-site.xml** configuration file of the node where the NodeManager instance resides:

Table 12-302 Parameter description

Parameter	Description	Default Value
mapreduce.shuffle.address	Indicates the specified address to run the shuffle service. The format is *IP:PORT*. The default value is empty. If this parameter is left empty, the local host IP address is bound. The default port number is 13562. **NOTE** If the value of *PORT* is different from that of **mapreduce.shuffle.port**, the **mapreduce.shuffle.port** value does not take effect.	-

12.18.6 Configuring the Cluster Administrator List

Scenario

This function is used to specify the MapReduce cluster administrator.

The administrator list is specified by **mapreduce.cluster.administrators**. The cluster administrator **admin** has all operation permissions.

Configuration

On the **All Configurations** page of the MapReduce service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-303 Parameter description

Parameter	Description	Default Value
mapreduce.cluster.acls.enabled	Indicates whether to enable permission control on Job History Server.	true
mapreduce.cluster.administrators	Indicates the administrator list of the MapReduce cluster. You can configure both users and user groups. Multiple users or user groups are separated by commas (,), and users and user groups are separated by spaces, for example, userA,userB groupA,groupB. The value * indicates all users or user groups.	For versions earlier than MRS 3.x: mapred For MRS 3.*x* or later: mapred supergroup,System_administrator_186

12.18.7 Introduction to MapReduce Logs

Log Description

Log paths:

- JobhistoryServer: **/var/log/Bigdata/mapreduce/jobhistory** (run log) and **/var/log/Bigdata/audit/mapreduce/jobhistory** (audit log)
- Container: **/srv/BigData/hadoop/data1/nm/containerlogs/application_${appid}/container_{$contid}**

NOTE

The logs of running tasks are stored in the preceding paths. After the running is complete, the system determines whether to aggregate the logs to an HDFS directory based on the YARN configuration. For details, see **Common YARN Parameters**.

Log archive rule:

The automatic compression and archive function is enabled for MapReduce logs. By default, a log file is automatically compressed when the size of the log file is greater than 50 MB. The name of the compressed log file is in the following format: *<Name of the original log>-<yyyy-mm-dd_hh-mm-ss>.[NO.]***.log.zip**. A maximum of 100 latest compressed files are reserved. The number of compressed files can be configured on the parameter configuration page.

In MapReduce, JobhistoryServer cleans the old log files stored in HDFS periodically. The default storage directory is **/mr-history/done**. **mapreduce.jobhistory.max-age-ms** is used to set the cleanup interval. The default value of this parameter is 1,296,000,000 ms, which indicates 15 days.

Table 12-304 MapReduce log list

Type	Name	Description
Run log	jhs-daemon-start-stop.log	Startup log file of the daemon process
	hadoop-<SSH_USER>-jhshadaemon-<hostname>.log	Run log file of the daemon process
	hadoop-<SSH_USER>-<process_name>-<hostname>.out	Log that records the MapReduce running environment information
	historyserver-<SSH_USER>-<DATE>-<PID>-gc.log	Log that records the garbage collection of the MapReduce service
	jhs-haCheck.log	Log that records the active and standby status of MapReduce instances

Type	Name	Description
	yarn-start-stop.log	Log that records the startup and stop of the MapReduce service
	yarn-prestart.log	Log that records cluster operations before the MapReduce service startup
	yarn-postinstall.log	Work log before the MapReduce service startup and after the installation
	yarn-cleanup.log	Log that records the cleanup logs about the uninstallation of the MapReduce service
	mapred-service-check.log	Log that records the health check details of the MapReduce service
	container_{$contid}	Container log
	hadoop-<SSH_USER>-<process_name>-<hostname>.log	MR run log
	mapred-switch-jhs.log	MR active/standby switchover log
	env.log	Environment information log before the instance is started or stopped
Audit log	mapred-audit-jobhistory.log	MapReduce operation audit log
	SecurityAuth.audit	MapReduce security audit log

Log Level

Table 12-305 describes the log levels supported by MapReduce The log levels are FATAL, ERROR, WARN, INFO, and DEBUG from high priority to low. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-305 Log level

Level	Description
FATAL	Logs of this level record critical error information about the current event processing.
ERROR	Logs of this level record error information about the current event processing.
WARN	Logs of this level record unexpected alarm information about the current event processing.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of the MapReduce service. For details, see **Modifying Cluster Service Configuration Parameters**.

Step 2 On the left menu bar, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

NOTE

The configurations take effect immediately without restarting the service.

----End

Log Format

The following table lists the MapReduce log formats.

Table 12-306 Log format

Type	Format	Example
Run log	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2020-01-26 14:18:59,109 | INFO | main | Client environment:java.compiler=<NA> | org.apache.zookeeper.Environment.logEnv(Environment.java:100)

Type	Format	Example
Audit log	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2020-01-26 14:24:43,605 | INFO | main-EventThread | USER=omm OPERATION=refreshAdminAcls TARGET=AdminService RESULT=SUCCESS | org.apache.hadoop.yarn.server.resourcemanager.RMAuditLogger$LogLevel $6.printLog(RMAuditLogger.java:91)

12.18.8 MapReduce Performance Tuning

12.18.8.1 Optimization Configuration for Multiple CPU Cores

Scenario

Optimization can be performed when the number of CPU cores is large, for example, the number of CPU cores is three times the number of disks.

Procedure

You can set the following parameters in either of the following ways:

- Configuration on the server:

  On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

- Configuration on the client:

  Modify the corresponding configuration file on the client.

  NOTE

  - Path of configuration files on the HDFS client: *Client installation directory*/**HDFS/hadoop/etc/hadoop/hdfs-site.xml**
  - Path of configuration files on the Yarn client: *Client installation directory*/**HDFS/hadoop/etc/hadoop/yarn-site.xml**.
  - Path of configuration files on the MapReduce client: *Client installation directory*/**HDFS/hadoop/etc/hadoop/mapred-site.xml**.

Table 12-307 Settings of multiple CPU cores

Configuration	Description	Parameter	Default Value	Server/ Client	Impact	Remarks
Number of slots in a node container	The combination of the following parameters determines the number of concurrent tasks (Map and Reduce tasks) of each node: • yarn.nodemanager.resource.memory-mb • mapreduce.map.memory.mb • mapreduce.reduce.memory.mb	yarn.nodemanager.resource.memory-mb **NOTE** For versions earlier than MRS 3.x: You need to configure this parameter on the MRS console. For MRS 3.*x* or later: You need to configure this parameter on FusionInsight Manager.	Versions earlier than MRS 3.*x*: 8192 MRS 3.*x* or later: 16384	Server	If data needs to be read from and written into disks for all tasks (Map/ Reduce tasks), a disk may be accessed by multiple processes at the same time, which leads to poor disk I/O performance. To ensure disk I/O performance, the number of concurrent access requests from a client to a disk cannot exceed 3.	The maximum number of concurrent containers must be [2.5 x Number of disks configured in Hadoop].
		mapreduce.map.memory.mb **NOTE** You need to set this parameter in the configuration file on the client in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/mapred-site.xml** path.	4096	Client		

Configuration	Description	Parameter	Default Value	Server/Client	Impact	Remarks
		mapreduce.reduce.memory.mb **NOTE** You need to set this parameter in the configuration file on the client in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/mapred-site.xml** path.	4096	Client		

Configuration	Description	Parameter	Default Value	Server/ Client	Impact	Remarks
Map output and compression	The Map task output before being written into disks can be compressed. This can save disk space, offer faster data write, and reduce the data traffic delivered to Reducer. You need to configure the following parameters on the client: ● **mapreduce.map.output.compress**: The Map task output can be compressed before it is transmitted over the network. It is a per-job	mapreduce.map.output.compress **NOTE** You need to set this parameter in the configuration file on the client in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/mapred-site.xml** path.	true	Client	The disk I/O is the bottleneck. Therefore, use a compression algorithm with a high compression rate.	Snappy is used. The benchmark test results show that Snappy delivers high performance and efficiency.
		mapreduce.map.output.compress.codec **NOTE** You need to set this parameter in the configuration file on the client in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/mapred-site.xml** path.	org.apache.hadoop.io.compress.Lz4Codec	Client		

Configuration	Description	Parameter	Default Value	Server/ Client	Impact	Remarks
	configuration. • **mapreduce.map.output.compress.codec**: the codec used for data compression					
Spills	mapreduce.map.sort.spill.percent	mapreduce.map.sort.spill.percent **NOTE** You need to set this parameter in the configuration file on the client in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/mapred-site.xml** path.	0.8	Client	Disk I/Os are the bottleneck. You can set the value of **mapreduce.task.io.sort.mb** to minimize the memory spilled to the disk.	-

Configuration	Description	Parameter	Default Value	Server/ Client	Impact	Remarks
Data packet size	When the HDFS client writes data to a data node, the data will be accumulated until a packet is generated. Then, the packet is transmitted over the network. **dfs.client-write-packet-size** specifies the data packet size. It can be specified by each job.	dfs.client-write-packet-size **NOTE** You need to set this parameter in the configuration file on the client in the *Client installation directory*/**HDFS/hadoop/etc/hadoop/hdfs-site.xml/** path.	262144	Client	The data node receives data packets from the HDFS client and writes data into disks through single threads. When disks are in the concurrent write state, increasing the data packet size can reduce the disk seek time and improve the I/O performance.	dfs.client-write-packet-size = 262144

12.18.8.2 Determining the Job Baseline

Scenario

The performance optimization effect is verified by comparing actual values with the baseline data. Therefore, determining optimal job baseline is critical to performance optimization.

When determining the job baseline, comply with the following rules:

- Making full use of cluster resources
- Setting the number of Map and Reduce tasks appropriately
- Setting the runtime of each task appropriately

Procedure

- **Rule 1: Making full use of cluster resources**

  Enable all nodes to handle tasks as actively as they can when a job is executed. Maximizing the number of concurrent tasks helps make full use of resources. You can achieve this purpose by adjusting the data volume to be processed and the number of Map and Reduce tasks.

  You can set **mapreduce.job.reduces** to control the number of Reduce tasks.

  The number of Map tasks depends on the InputFormat type and whether the data file to be processed can be split. By default, TextFileInputFormat allocates Map tasks based on the number of blocks, that is, one Map task for each block. You can adjust the following parameters to improve resource utilization.

  Parameter portal:

  On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

Parameter	Description	Default Value
mapreduce.input.fileinputformat.split.maxsize	Indicates the maximum size of the data block into which the Map input information is to be split. The shard size can be calculated based on its size customized by the user and the block size of each file. The formula is as follows: `splitSize = Math.max(minSize, Math.min(maxSize, blockSize))` If **maxSize** is bigger than **blockSize**, a block is a shard. If **maxSize** is smaller than **blockSize**, a block will be split into multiple shards. If the size of the remaining data in a block is smaller than **splitSize**, the remaining data will be treated as a separated shard.	-
mapreduce.input.fileinputformat.split.minsize	Indicates the minimum size of a data shard.	0

- **Principle 2: Setting Reduce tasks to be executed in one round.**

  Avoid the following scenarios:

  – Most of Reduce tasks are completed in the first round, but there is still one Reduce task left running. The execution of the last Reduce task extends the runtime of the job. Therefore, reduce the number of Reduce tasks to enable all of them to run at the same time.

  - All Map tasks are completed, but there are still Reduce tasks running on some nodes. In this case, the cluster resources are not fully utilized. You need to increase the number of Reduce tasks to enable each node to handle tasks.

- **Rule 3: Setting the runtime of each task appropriately**

  If each Map or Reduce task of a job takes only a few seconds, most time of the job is wasted on scheduling tasks and starting and stopping processes. Therefore, you need to increase the data volume to be processed in each task. The preferred processing time for each task is 1 minute.

  You can configure the following parameters to adjust the processing time in a task.

  Parameter portal:

  On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

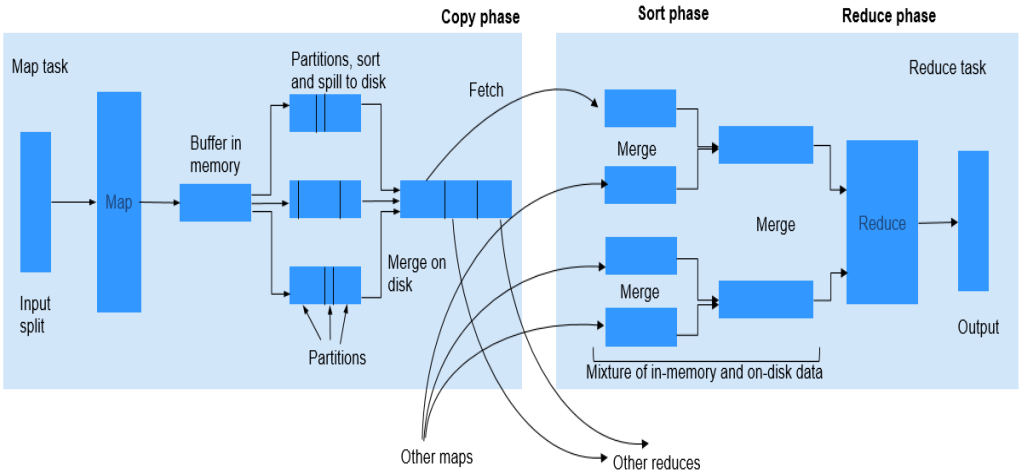
Parameter	Description	Default Value
mapreduce.input.fileinputformat.split.maxsize	Indicates the maximum size of the data block into which the Map input information is to be split. The shard size can be calculated based on its size customized by the user and the block size of each file. The formula is as follows: `splitSize = Math.max(minSize, Math.min(maxSize, blockSize))` If **maxSize** is bigger than **blockSize**, a block is a shard. If **maxSize** is smaller than **blockSize**, a block will be split into multiple shards. If the size of the remaining data in a block is smaller than **splitSize**, the remaining data will be treated as a separated shard.	-
mapreduce.input.fileinputformat.split.minsize	Indicates the minimum size of a data shard.	0

12.18.8.3 Streamlining Shuffle

Scenario

During the shuffle procedure of MapReduce, the Map task writes intermediate data into disks, and the Reduce task copies and adds the data to the reduce function. Hadoop provides lots of parameters for the optimization.

Figure 12-37 Shuffle process



Procedure

1. **Improving Performance in Map Phase**
   - Determine the memory used by Map.

     To determine whether Map has sufficient memory, check the number of GCs and the ratio of the GC time over the total task time in counters of completed jobs. Normally, the GC time cannot exceed 10% of the task time (that is, GC time elapsed (ms)/CPU time spent (ms) < 10%).

     You can improve Map performance by adjusting the following parameters.

     Parameter portal:

     On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

     Table 12-308 Parameter description

Parameter	Description	Default Value
mapreduce.map.memory.mb	Memory restriction of a Map task.	4096

Parameter	Description	Default Value
mapreduce.map.java.opts	JVM parameter of the Map subtask. If this parameter is set, it will replace the **mapred.child.java.opts** parameter. If **-Xmx** is not set, the value of **Xmx** is calculated based on **mapreduce.map.memory.mb** and **mapreduce.job.heap.memory-mb.ratio**.	For versions earlier than MRS 3.*x*: -Xmx2048M -Djava.net.preferIPv4Stack=true For MRS cluster 3.*x* and later versions: ● Clusters with Kerberos authentication enabled: -Djava.net.preferIPv4Stack=true -Djava.net.preferIPv6Addresses=false -Djava.security.krb5.conf=*${BIGDATA_HOME}*/common/runtime/krb5.conf -Dbeetle.application.home.path=*${BIGDATA_HOME}*/common/runtime/security/config ● Clusters with Kerberos authentication disabled: -Djava.net.preferIPv4Stack=true -Djava.net.preferIPv6Addresses=false -Dbeetle.application.home.path=*${BIGDATA_HOME}*/common/runtime/security/config

It is recommended that the **-Xmx** in **mapreduce.map.java.opts** is 0.8 times the value of **mapreduce.map.memory.mb**.

– Using Combiner

Combiner is an optional procedure in the Map phase, in which the intermediate results with the same key value are combined. Generally, set

the reduce class to combiner. Combiner helps reduce the intermediate result output of Map, thereby consuming less network bandwidth during the shuffle process. You can use the following API to set a combiner class for a specific job.

Table 12-309 Combiner API

Class	API	Description
org.apache.hadoop.mapreduce.Job	public void setCombinerClass(Class<? extends Reducer> cls)	API used to set a combiner class for a specific job.

2. **Improving Performance in Copy Phase**
   - Compress data.

     Compress the intermediate output of Map. Data compression reduces the data to be transferred over the network. However, data compression and decompression consume more CPU. Determine whether to compress the intermediate results of Map based on site requirements. If a task is bandwidth-intensive, data compression improves processing performance. As for the bulkload optimization, compression of the intermediate output improves the performance by 60%.

     To improve copy performance, set **mapreduce.map.output.compress** to **true** and **mapreduce.map.output.compress.codec** to **org.apache.hadoop.io.compress.SnappyCodec**.
3. **Improving Performance in Merge Phase**

   To improve merge performance, configure the following parameters to reduce the number of times that Reduce writes data to disks.

   Parameter portal:

   On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-310 Parameter description

Parameter	Description	Default Value
mapreduce.reduce.merge.inmem.threshold	Threshold of the number of files for the in-memory merge process. When the accumulated number of files reaches the threshold, the process of in-memory merge and spilling to disks is initiated. If the value is less than or equal to **0**, the threshold does not take effect and the merge is triggered only based on the RAMFS memory usage.	1000
mapreduce.reduce.shuffle.merge.percent	Usage threshold for initiating in-memory merge, indicating the percentage of memory allocated to the Map outputs (defined by **mapreduce.reduce.shuffle.input.buffer.percent**).	0.66
mapreduce.reduce.shuffle.input.buffer.percent	Percentage of memory to be allocated from the maximum heap size to storing Map outputs during the Shuffle.	0.70
mapreduce.reduce.input.buffer.percent	Percentage of memory (relative to the maximum heap size) to retain Map outputs during the Reduce. When the Shuffle is completed, all remaining Map outputs in memory must use less than this threshold before the Reduce begins.	0.0

12.18.8.4 AM Optimization for Big Tasks

Scenario

A big job containing 100,000 Map tasks fails. It is found that the failure is triggered by the slow response of ApplicationMaster (AM).

When the number of tasks increases, the number of objects managed by the AM increases, which requires much more memory for management. The default memory heap for AM is 1 GB.

Procedure

You can improve the AM performance by setting the following parameters.

Navigation path for setting parameters:

Adjust the following parameters in the **mapred-site.xml** configuration file on the client to adjust the following parameters: The **mapred-site.xml** configuration file is in the **conf** directory of the client installation path, for example, **/opt/client/Yarn/config**.

Parameter	Description	Default Value
yarn.app.mapreduce.am.resource.mb	This parameter must be greater than the heap size specified by **yarn.app.mapreduce.am.command-opts**. Unit: MB	1536
yarn.app.mapreduce.am.command-opts	Indicates the JVM startup parameters loaded to MapReduce ApplicationMaster.	For versions earlier than MRS 3.x: -Xmx1024m -XX:CMSFullGCsBeforeCompaction=1 -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:+UseCMSCompactAtFullCollection -verbose:gc MRS 3.x or later: -Xmx1024m -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -verbose:gc -Djava.security.krb5.conf=${KRB5_CONFIG} -Dhadoop.home.dir=*${BIGDATA_HOME}*/FusionInsight_HD_*xxx*/install/FusionInsight-Hadoop-*xxx*/hadoop

12.18.8.5 Speculative Execution

Scenario

If a cluster has hundreds or thousands of nodes, the hardware or software fault of a node may prolong the execution time of the entire task (as most tasks are already completed, the system is still waiting for the task running on the faulty node). Speculative execution allows a task to be executed on multiple machines. You can disable speculative execution for small clusters.

Procedure

Navigation path for setting parameters:

On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

Parameter	Description	Default Value
mapreduce.map.speculative	Sets whether to execute multiple instances of some map tasks concurrently. **true** indicates that speculative execution is enabled.	false
mapreduce.reduce.speculative	Sets whether to execute multiple instances of some reduce tasks concurrently. **true** indicates that speculative execution is enabled.	false

12.18.8.6 Using Slow Start

Scenario

The Slow Start feature specifies the proportion of Map tasks to be completed before Reduce tasks are started. If the Reduce tasks are started too early, resources will be occupied, thereby reducing task running efficiency. However, if the Reduce tasks are started at an appropriate time, resource usage during shuffle and task running efficiency will be improved. For example, the MapReduce job includes 15 Map tasks and a cluster can start 10 Map tasks, there are 5 Map tasks remained after a round of Map tasks is completed and the cluster has available resources. In this case, you can configure the value of Slow Start to a value less than 1 (for example, 0.8), then the Reduce tasks can make use of the remaining cluster resources.

Procedure

Parameter portal:

On the **All Configurations** page of the MapReduce service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

Parameter	Description	Default Value
mapreduce.job.reduce.slowstart.completedmaps	Fraction of the number of Maps in the job which should be completed before Reduces are scheduled for the job. By default, the Reduce tasks start when all the Map tasks are completed.	1.0

12.18.8.7 Optimizing Performance for Committing MR Jobs

Scenario

By default, if an MR job generates a large number of output files, it takes a long time for the job to commit the temporary outputs of a task to the final output

directory in the commit phase. In large clusters, the time-consuming commit process of jobs greatly affects the performance.

In this case, you can set the **mapreduce.fileoutputcommitter.algorithm.version** to **2** to improve the performance in the commit phase of MR jobs.

Procedure

Navigation path for setting parameters:

On the **All Configurations** page of the Yarn service, enter a parameter name in the search box. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-311 Parameter description

Parameter	Description	Default Value
mapreduce.fileoutputcommitter.algorithm.version	Indicates the algorithm version submitted by a job. The value is **1** or **2**. **NOTE** **2** is the recommended algorithm version. This algorithm enables tasks to directly commit the output results of each task to the final result output directory, reducing the time for the results of large jobs are committed.	2

12.18.9 Common Issues About MapReduce

12.18.9.1 Why Does It Take a Long Time to Run a Task Upon ResourceManager Active/Standby Switchover?

Question

MapReduce job takes a very long time (more than 10minutes) when the ResourceManager switch while the job is running.

Answer

This is because, ResorceManager HA is enabled but the ResourceManager work preserving restart is not enabled.

If ResorceManager work preserving restart is not enabled, then ResorceManager switch containers are killed which causes the ResorceManager to timeout the ApplicationMaster. For ResorceManager work preserving restart feature details, see **http://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-site/ResourceManagerRestart.html**.

The following method can be used to solve the issue:

Enable the ResorceManager work preserving restart feature by configuring the following parameter.

yarn.resourcemanager.work-preserving-recovery.enabled=**true**

12.18.9.2 Why Does a MapReduce Task Stay Unchanged for a Long Time?

Question

MapReduce job is not progressing for long time

Answer

This is because of less memory. When the memory is less, the time taken by the job to copy the map output increases significantly.

In order to reduce the waiting time, increase the heap memory.

The job configuration should be tuned according to number of mappers and data size processed by each mapper. Based on the input data size, tune the following configurations accordingly for feasible performance.

- **mapreduce.reduce.memory.mb**
- **mapreduce.reduce.java.opts**

Example: If the data size is 5 GB with 10 mappers, then the ideal heap memory would be 1.5 GB. Increase the heap memory size according with the increase in data size.

12.18.9.3 Why the Client Hangs During Job Running?

Question

Why is the client unavailable when the MR ApplicationMaster or ResourceManager is moved to the D state during job running?

Answer

When a task is running, the MR ApplicationMaster or ResourceManager is moved to D state (uninterrupted sleep state) or T state (stopped state). The client waits to return the task running state, but the MR ApplicationMaster does not return. Therefore, the client remains in the waiting state.

To avoid the preceding scenario, use the **ipc.client.rpc.timeout** configuration item in the **core-site.xml** file to set the client timeout interval.

The value of this parameter is millisecond. The default value is **0**, indicating that no timeout occurs. The client timeout interval ranges from 0 ms to 2,147,483,647 ms.

NOTE

- If the Hadoop process is in the D state, restart the node where the process is located.
- The **core-site.xml** configuration file is stored in the **conf** directory of the client installation path, for example, **/opt/hadoopClient/Yarn/config**.

12.18.9.4 Why Cannot HDFS_DELEGATION_TOKEN Be Found in the Cache?

Question

In security mode, why delegation token HDFS_DELEGATION_TOKEN is not found in the cache?

Answer

In MapReduce, by default HDFS_DELEGATION_TOKEN will be canceled after the job completion. So if the token has to be re- used for the next job then the token will not be found in the cache.

To re-use the same token in subsequent job set the below parameter for the MR job configuration. When it is false the user can re-sue the same token.

```
jobConf.setBoolean("mapreduce.job.complete.cancel.delegation.tokens", false);
```

12.18.9.5 How Do I Set the Task Priority When Submitting a MapReduce Task?

Question

How do I set the job priority when submitting a MapReduce task?

Answer

You can add the parameter **-Dmapreduce.job.priority=<priority>** in the command to set task priority when submitting MapReduce tasks on the client. The format is as follows:

yarn jar *<jar> [mainClass]* ***-Dmapreduce.job.priority=****<priority> [path1] [path2]*

The parameters in the command are described as follows:

- *<jar>*: specifies the name of the JAR package to be run.
- *[mainClass]*: specifies the **main** method of the class for an application project in a JAR file.
- *<priority>*: specifies the priority of a task. The value can be **VERY_HIGH**, **HIGH**, **NORMAL**, **LOW**, or **VERY_LOW**.
- *[path1]*: specifies the data input path.
- *[path2]*: specifies the data output path.

For example, set the **/opt/client/HDFS/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples*.jar** package to a high-priority task.

yarn jar /opt/client/HDFS/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples*.jar wordcount -Dmapreduce.job.priority=VERY_HIGH /DATA.txt /out/

12.18.9.6 Why Physical Memory Overflow Occurs If a MapReduce Task Fails?

Question

The HBase bulkload task has 210,000 Map tasks and 10,000 Reduce tasks. The MapReduce task fails to be executed, and the physical memory of ApplicationMaster overflows.

```
For more detailed output, check the application tracking page:https://bigdata-55:8090/cluster/app/
application_1449841777199_0003
Then click on links to logs of each attempt.
Diagnostics: Container [pid=21557,containerID=container_1449841777199_0003_02_000001] is running
beyond physical memory limits
Current usage: 1.0 GB of 1 GB physical memory used; 3.6 GB of 5 GB virtual memory used. Killing container.
Dump of the process-tree for container_1449841777199_0003_02_000001 :
|- PID PPID PGRPID SESSID CMD_NAME USER_MODE_TIME(MILLIS) SYSTEM_TIME(MILLIS)
VMEM_USAGE(BYTES) RSSMEM_USAGE(PAGES) FULL_CMD_LINE
|- 21584 21557 21557 21557 (java) 12342 1627 3871748096 271331 ${BIGDATA_HOME}/jdk1.8.0_51//bin/
java
-Djava.io.tmpdir=/srv/BigData/hadoop/data1/nm/localdir/usercache/hbase/appcache/
application_1449841777199_0003/container_1449841777199_0003_02_000001/tmp -
Dlog4j.configuration=container-log4j.properties
-Dyarn.app.container.log.dir=/srv/BigData/hadoop/data1/nm/containerlogs/
application_1449841777199_0003/container_1449841777199_0003_02_000001 -
Dyarn.app.container.log.filesize=0 -Dhadoop.root.logger=INFO,CLA
-Dhadoop.root.logfile=syslog -Xmx784m org.apache.hadoop.mapreduce.v2.app.MRAppMaster
|- 21557 21547 21557 21557 (bash) 0 0 13074432 368 /bin/bash -c ${BIGDATA_HOME}/jdk1.8.0_51//bin/
java
-Djava.io.tmpdir=/srv/BigData/hadoop/data1/nm/localdir/usercache/hbase/appcache/
application_1449841777199_0003/container_1449841777199_0003_02_000001/tmp -
Dlog4j.configuration=container-log4j.properties
-Dyarn.app.container.log.dir=/srv/BigData/hadoop/data1/nm/containerlogs/
application_1449841777199_0003/container_1449841777199_0003_02_000001 -
Dyarn.app.container.log.filesize=0 -Dhadoop.root.logger=INFO,CLA
-Dhadoop.root.logfile=syslog -Xmx784m org.apache.hadoop.mapreduce.v2.app.MRAppMaster 1>/srv/
BigData/hadoop/data1/nm/containerlogs/application_1449841777199_0003/
container_1449841777199_0003_02_000001/stdout
2>/srv/BigData/hadoop/data1/nm/containerlogs/application_1449841777199_0003/
container_1449841777199_0003_02_000001/stderr
Container killed on request. Exit code is 143
Container exited with a non-zero exit code 143
Failing this attempt. Failing the application.
```

Answer

This is a performance specification problem. The root cause of the MapReduce task execution failure is the memory overflow of ApplicationMaster, that is, the NodeManager kills the task due to the physical memory overflow.

Solutions:

Increase the memory of ApplicationMaster and optimize the following parameters in the **mapred-site.xml** configuration file on the client:

- **yarn.app.mapreduce.am.resource.mb**
- **yarn.app.mapreduce.am.command-opts**. The recommended value of **-Xmx** is 0.8 x **yarn.app.mapreduce.am.resource.mb**.

Specification:

ApplicationMaster supports 24,000 concurrent containers when the configuration is as follows:

- **yarn.app.mapreduce.am.resource.mb**=2048
- In **yarn.app.mapreduce.am.command-opts**, **-Xmx** is **1638m**.

12.18.9.7 After the Address of MapReduce JobHistoryServer Is Changed, Why the Wrong Page is Displayed When I Click the Tracking URL on the ResourceManager WebUI?

Question

After the address of MapReduce JobHistoryServer is changed, why the wrong page is displayed when I click the tracking URL on the ResourceManager WebUI?

Answer

JobHistoryServer address (mapreduce.jobhistory.address / mapreduce.jobhistory.webapp.<https.>address) is the parameter of MapReduce. The MapReduce client will submit the address together with jobs to ResourceManager. After ResourceManager completing the jobs, the parameter is saved in RMStateStore as the target address for viewing history job information.

If the JobHistoryServer address is changed, update the address in the configuration file of the MapReduce client in time. If the address is not updated, the page of earlier JobHistoryServer is displayed when you click the tracking URL of the new job. The target address of information about MapReduce jobs running before the change of address cannot be changed, so the wrong page is also displayed when you click the tracking URL. You can check the history information by accessing the new JobHistoryServer address.

12.18.9.8 MapReduce Job Failed in Multiple NameService Environment

Question

MapReduce or Yarn job fails in multiple nameService environment using viewFS.

Answer

When using viewFS only the mount directories are accessible, so the most possible cause is that the path configured is not in one of the mounted paths. For example:

```
<property>
<name>fs.defaultFS</name>
<value>viewfs://ClusterX/</value>
</property>
<property>
<name>fs.viewfs.mounttable.ClusterX.link./folder1</name>
<value>hdfs://NS1/folder1</value>
</property>
<property>
<name>fs.viewfs.mounttable.ClusterX.link./folder2</name>
<value>hdfs://NS2/folder2</value>
</property>
```

For all the MR properties which depends on HDFS, should use the paths inside mount folders.

Incorrect:

```
<property>
<name>yarn.app.mapreduce.am.staging-dir</name>
<value>/tmp/hadoop-yarn/staging</value>
</property>
```

As the root folder (/) is not accessible in viewFS.

Correct:

```
<property>
<name>yarn.app.mapreduce.am.staging-dir</name>
<value>/folder1/tmp/hadoop-yarn/staging</value>
</property>
```

12.18.9.9 Why a Fault MapReduce Node Is Not Blacklisted?

Question

MapReduce task fails and the ratio of fault nodes to all nodes is smaller than the blacklist threshold configured by **yarn.resourcemanager.am-scheduling.node-blacklisting-disable-threshold**. Why the fault node not be blacklisted?

Answer

If the blacklisted percentage exceeds the threshold, all blacklisted nodes are released. Traditionally, the blacklist percentage is the ratio of fault nodes to all nodes in the cluster. Currently, each node has a label expression. Therefore, the blacklist percentage needs to be calculated based on the number of nodes related to valid node label expressions. In other way, the blacklist percentage is the ratio of fault nodes related to valid node label expressions.

Assume that there are 100 nodes in the cluster, including 10 nodes (labelA) related to valid node label expressions. Assume that all nodes related to valid node label expressions are faulty and default blacklist threshold is 0.33. In traditional calculation method, 10/100 = 0.1, which is far smaller than the threshold (0.33). In this case, the 10 nodes will never get released. Therefore, MapReduce always cannot obtain nodes and applications cannot run properly. In practice, the blacklist percentage needs to be calculated based on the total number of nodes related to valid node label expressions: 10/10 = 1 is greater than the blacklist threshold and all nodes are released.

Therefore, even the ratio of fault nodes to all nodes in the cluster is below the threshold, all nodes in the blacklist are released.

12.19 Using Oozie

12.19.1 Using Oozie from Scratch

Oozie is an open-source workflow engine that is used to schedule and coordinate Hadoop jobs.

Oozie can be used to submit a wide array of jobs, such as Hive, Spark2x, Loader, MapReduce, Java, DistCp, Shell, HDFS, SSH, SubWorkflow, Streaming, and scheduled jobs.

This section describes how to use the Oozie client to submit a MapReduce job.

Prerequisites

The client has been installed. For example, the installation directory is **/opt/client**. The client directory in the following operations is only an example. Change it based on the actual installation directory onsite.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory, for example, **/opt/Bigdata/client**:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Check the cluster authentication mode.

- If the cluster is in security mode, run the following command to authenticate the user: *UserOozie* indicates the user who submits tasks.

  kinit *UserOozie*

- If the cluster is in normal mode, go to **Step 5**.

Step 5 Upload the Oozie configuration file and JAR package to HDFS.

hdfs dfs -mkdir /user/*UserOozie*

hdfs dfs -put -f /opt/client/Oozie/oozie-client-*/examples /user/*UserOozie/*

NOTE

- **/opt/client/** is an example client installation directory. Change it to the actual installation directory.
- **UserOozie** indicates the name of the user who submits jobs.

Step 6 Run the following commands to modify the job execution configuration file:

cd /opt/client/Oozie/oozie-client-*/examples/apps/map-reduce/

vi job.properties

```
nameNode=hdfs://hacluster
resourceManager=10.64.35.161:8032 (10.64.35.161 is the service plane IP address of the Yarn resourceManager (active) node, and 8032 is the port number of yarn.resourcemanager.port)
queueName=default
examplesRoot=examples
user.name=admin
oozie.wf.application.path=${nameNode}/user/${user.name}/${examplesRoot}/apps/map-reduce# HDFS upload path
outputDir=map-reduce
oozie.wf.rerun.failnodes=true
```

Step 7 Run the following command to execute the Oozie job:

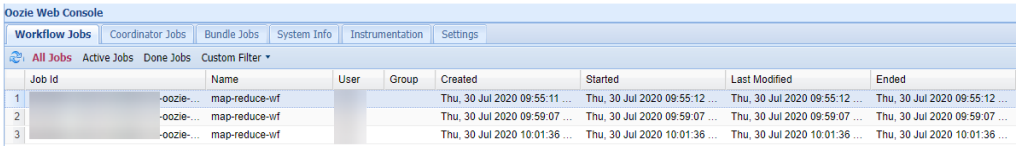
oozie job -oozie https://*Host name of the Oozie role*:21003/oozie/ -config job.properties -run

```
[root@kwephispra44947 map-reduce]# oozie job -oozie https://kwephispra44948:21003/oozie/ -config job.properties -run
.....
job: 0000000-200730163829770-oozie-omm-W
```

Step 8 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Step 9 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Oozie**, click the hyperlink next to **Oozie WebUI** to go to the Oozie page, and view the task execution result on the Oozie web UI.

Figure 12-38 Task execution result



----End

12.19.2 Using the Oozie Client

Scenario

This section describes how to use the Oozie client in an O&M scenario or service scenario.

Prerequisites

- The client has been installed. For example, the installation directory is **/opt/client**. The client directory in the following operations is only an example.
- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login.

Using the Oozie Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to switch to the client installation directory (change it to the actual installation directory):

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Check the cluster authentication mode.

- If the cluster is in security mode, run the following command to authenticate the user: *exampleUser* indicates the name of the user who submits tasks.

kinit *exampleUser*

- If the cluster is in normal mode, go to **Step 5**.

Step 5 Perform the following operations to configure Hue:

1. Configure the Spark2x environment (skip this step if the Spark2x task is not involved):

   hdfs dfs -put /opt/client/Spark2x/spark/jars/*.jar /user/oozie/share/lib/spark2x/

   When the JAR package in the HDFS directory **/user/oozie/share** changes, you need to restart the Oozie service.

2. Upload the Oozie configuration file and JAR package to HDFS.

   hdfs dfs -mkdir /user/*exampleUser*

   hdfs dfs -put -f /opt/client/Oozie/oozie-client-*/examples /user/*exampleUser*/

   NOTE

   - *exampleUser* indicates the name of the user who submits tasks.
   - If the user who submits the task and other files except **job.properties** are not changed, client installation directory **Oozie/oozie-client-*/examples** can be repeatedly used after being uploaded to HDFS.
   - Resolve the JAR file conflict between Spark and Yarn about Jetty.

     hdfs dfs -rm -f /user/oozie/share/lib/spark/jetty-all-9.2.22.v20170606.jar
   - In normal mode, if **Permission denied** is displayed during the upload, run the following commands:

     su - omm

     source /opt/client/bigdata_env

     hdfs dfs -chmod -R 777 /user/oozie

     exit

----End

12.19.3 Using Oozie Client to Submit an Oozie Job

12.19.3.1 Submitting a Hive Job

Scenario

This section describes how to use the Oozie client to submit a Hive job.

Hive jobs are divided into the following types:

- Hive job

  Hive job that is connected in JDBC mode
- Hive2 job

  Hive job that is connected in Beeline mode

This section describes how to submit a Hive job using the Oozie client.

NOTE

- The procedure for submitting a Hive2 job using the Oozie client is the same as that for submitting a Hive job. You only need to change **/Hive** in the procedure to **/Hive2**.

  For example, if the Hive job running directory is **/opt/client/Oozie/oozie-client-*/examples/apps/hive/**, then the running directory of Hive2 is **/opt/client/Oozie/oozie-client-*/examples/apps/hive2/**.
- You are advised to download the latest client.

Prerequisites

- The Hive and Oozie components and clients have been installed and are running properly.
- You have created or obtained the human-machine account and password for accessing the Oozie service.

  NOTE

  - This user must belong to the **hadoop**, **supergroup**, and **hive** groups and be assigned with the Oozie role operation permission. If the multi-instance function is enabled for Hive, the user must belong to a specific Hive instance group, for example, **hive3**.
  - This user must also be assigned the **manager_viewer** role at least.
- You have obtained the URL of the Oozie server (any instance) in the running state, for example, **https://10.1.130.10:21003/oozie**.
- You have obtained the name of the Oozie server, for example, **10-1-130-10**.
- You have obtained the IP address of the active Yarn ResourceManager, for example, **10.1.130.11**.

Procedure

Step 1 Log in to the node where the Oozie client is installed as the client installation user.

Step 2 Run the following command to obtain the installation environment. **/opt/client/** is an example client installation path.

source /opt/client/bigdata_env

Step 3 Check the cluster authentication mode.

- If the cluster is in security mode, run the **kinit** command to authenticate users.

  For example, the **oozieuser** user is authenticated using the following command:

  kinit oozieuser
- If the cluster is in normal mode, go to **Step 4**.

Step 4 Run the following command to go to the example directory:

cd /opt/client/Oozie/oozie-client-*/examples/apps/hive/

Table 12-312 lists the files that you need to pay attention to in the directory.

Table 12-312 File description

File	Description
hive-site.xml	Configuration file of a Hive job
job.properties	Parameter definition file of a workflow
script.q	SQL script of a Hive job
workflow.xml	Rule definition file of a workflow

Step 5 Run the following command to edit the **job.properties** file:

vi job.properties

Perform the following modifications:

Change the value of **userName** to the name of the human-machine user who submits the job, for example, **userName=oozieuser**.

Step 6 Run the **oozie job** command to run the workflow file:

oozie job -oozie https://*Host name of the Oozie role*:**21003/oozie/ -config job.properties -run**

NOTE

- The command parameters are described as follows:
  -oozie URL of the Oozie server that executes a job
  -config Workflow property file
  -run Executing a workflow
- If a job ID, for example, **job: 0000021-140222101051722-oozie-omm-W**, is displayed after the workflow file is executed, the job is successfully submitted. You can view the execution results on the Oozie management page.
  Log in to the Oozie web UI at **https**://*IP address of the Oozie role*:**21003/oozie** as user **oozieuser**.
  On the Oozie web UI, you can view the submitted workflow information based on the job ID in the table on the page.

----End

12.19.3.2 Submitting a Spark2x Job

Scenario

This section describes how to submit a Spark2x job using the Oozie client.

NOTE

You are advised to download the latest client.

Prerequisites

- The Spark2x and Oozie components and clients have been installed and are running properly.

If the current client is an earlier version, you need to download and install the client again.

- You have created or obtained the human-machine account and password for accessing the Oozie service.

  NOTE

  - This user must belong to the **hadoop**, **supergroup**, and **hive** groups and be assigned with the Oozie role operation permission. If the multi-instance function is enabled for Hive, the user must belong to a specific Hive instance group, for example, **hive3**.
  - This user must also be assigned the **manager_viewer** role at least.

- You have obtained the URL of the Oozie server (any instance) in the running state, for example, **https://10.1.130.10:21003/oozie**.
- You have obtained the name of the Oozie server, for example, **10-1-130-10**.
- You have obtained the IP address of the active Yarn ResourceManager, for example, **10.1.130.11**.

Procedure

Step 1 Log in to the node where the Oozie client is installed as the client installation user.

Step 2 Run the following command to obtain the installation environment. **/opt/client/** is an example client installation path.

source /opt/client/bigdata_env

Step 3 Check the cluster authentication mode.

- If the cluster is in security mode, run the **kinit** command to authenticate users.

  For example, the **oozieuser** user is authenticated using the following command:

  kinit oozieuser

- If the cluster is in normal mode, go to **Step 4**.

Step 4 Run the following command to go to the example directory:

cd /opt/client/Oozie/oozie-client-*/examples/apps/spark2x/

Table 12-313 lists the files that you need to pay attention to in the directory.

Table 12-313 File description

File	Description
job.properties	Parameter definition file of a workflow
workflow.xml	Rule definition file of a workflow
lib	Directory of the JAR file on which a workflow depends

Step 5 Run the following command to edit the **job.properties** file:

vi job.properties

Perform the following modifications:

Change the value of **userName** to the name of the human-machine user who submits the job, for example, **userName=oozieuser**.

Step 6 Run the **oozie job** command to run the workflow file:

oozie job -oozie https://*Host name of the Oozie role*:**21003/oozie/ -config job.properties -run**

NOTE

- The command parameters are described as follows:

  -oozie URL of the Oozie server that executes a job

  -config Workflow property file

  -run Executing a workflow

- If a job ID, for example, **job: 0000021-140222101051722-oozie-omm-W**, is displayed after the workflow file is executed, the job is successfully submitted. You can view the execution results on the Oozie management page.

  Log in to the Oozie web UI at **https**://*IP address of the Oozie role*:**21003/oozie** as user **oozieuser**.

  On the Oozie web UI, you can view the submitted workflow information based on the job ID in the table on the page.

----End

12.19.3.3 Submitting a Loader Job

Scenario

This section describes how to submit a Loader job using the Oozie client.

NOTE

You are advised to download the latest client.

Prerequisites

- The Hive and Oozie components and clients have been installed and are running properly.
- You have created or obtained the human-machine account and password for accessing the Oozie service.

  NOTE

  - This user must belong to the **hadoop**, **supergroup**, and **hive** groups and be assigned with the Oozie role operation permission. If the multi-instance function is enabled for Hive, the user must belong to a specific Hive instance group, for example, **hive3**.
  - This user must also be assigned the **manager_viewer** role at least.
- You have obtained the URL of the Oozie server (any instance) in the running state, for example, **https://10.1.130.10:21003/oozie**.

- You have obtained the name of the Oozie server, for example, **10-1-130-10**.
- You have obtained the IP address of the active Yarn ResourceManager, for example, **10.1.130.11**.
- You have created a Loader job to be scheduled and obtained the job ID.

Procedure

Step 1 Log in to the node where the Oozie client is installed as the client installation user.

Step 2 Run the following command to obtain the installation environment. **/opt/client/** is an example client installation path.

source /opt/client/bigdata_env

Step 3 Check the cluster authentication mode.

- If the cluster is in security mode, run the **kinit** command to authenticate users.

  For example, the **oozieuser** user is authenticated using the following command:

  kinit oozieuser
- If the cluster is in normal mode, go to **Step 4**.

Step 4 Run the following command to go to the example directory:

cd /opt/client/Oozie/oozie-client-*/examples/apps/sqoop/

Table 12-314 lists the files that you need to pay attention to in the directory.

Table 12-314 File description

File	Description
job.properties	Parameter definition file of a workflow
workflow.xml	Rule definition file of a workflow

Step 5 Run the following command to edit the **job.properties** file:

vi job.properties

Perform the following modifications:

Change the value of **userName** to the name of the human-machine user who submits the job, for example, **userName=oozieuser**.

Step 6 Run the following command to edit the **workflow.xml** file:

vi workflow.xml

Perform the following modifications:

Change the value of **command** to the ID of the Loader job to be scheduled, for example, **1**.

Upload the **workflow.xml** file to the HDFS path in the **job.properties** file.

hdfs dfs -put -f workflow.xml /user/userName/examples/apps/sqoop

Step 7 Run the **oozie job** command to run the workflow file:

oozie job -oozie https://*Host name of the Oozie role*:21003/oozie/ -config job.properties -run

NOTE

- The command parameters are described as follows:

  -oozie URL of the Oozie server that executes a job

  -config Workflow property file

  -run Executing a workflow
- If a job ID, for example, **job: 0000021-140222101051722-oozie-omm-W**, is displayed after the workflow file is executed, the job is successfully submitted. You can view the execution results on the Oozie management page.

  Log in to the Oozie web UI at **https**://*IP address of the Oozie role*:**21003/oozie** as user **oozieuser**.

  On the Oozie web UI, you can view the submitted workflow information based on the job ID in the table on the page.

----End

12.19.3.4 Submitting a DistCp Job

Scenario

This section describes how to submit a DistCp job using the Oozie client.

NOTE

You are advised to download the latest client.

Prerequisites

- The HDFS and Oozie components and clients have been installed and are running properly.

  If the current client is an earlier version, you need to download and install the client again.
- You have created or obtained the human-machine account and password for accessing the Oozie service.

  NOTE

  - This user must belong to the **hadoop**, **supergroup**, and **hive** groups and be assigned with the Oozie role operation permission. If the multi-instance function is enabled for Hive, the user must belong to a specific Hive instance group, for example, **hive3**.
  - This user must also be assigned the **manager_viewer** role at least.
- You have obtained the URL of the Oozie server (any instance) in the running state, for example, **https://10.1.130.10:21003/oozie**.
- You have obtained the name of the Oozie server, for example, **10-1-130-10**.
- You have obtained the IP address of the active Yarn ResourceManager, for example, **10.1.130.11**.

Procedure

Step 1 Log in to the node where the Oozie client is installed as the client installation user .

Step 2 Run the following command to obtain the installation environment. **/opt/client/** is an example client installation path.

source /opt/client/bigdata_env

Step 3 Check the cluster authentication mode.

- If the cluster is in security mode, run the **kinit** command to authenticate users.

  For example, the **oozieuser** user is authenticated using the following command:

  kinit oozieuser
- If the cluster is in normal mode, go to **Step 4**.

Step 4 Run the following command to go to the example directory:

cd /opt/client/Oozie/oozie-client-*/examples/apps/distcp/

Table 12-315 lists the files that you need to pay attention to in the directory.

Table 12-315 File description

File	Description
job.properties	Parameter definition file of a workflow
workflow.xml	Rule definition file of a workflow

Step 5 Run the following command to edit the **job.properties** file:

vi job.properties

Perform the following modifications:

Change the value of **userName** to the name of the human-machine user who submits the job, for example, **userName=oozieuser**.

Step 6 Whether DistCp is not deployed across security clusters.

- If yes, go to **Step 7**.
- If no, go to **Step 9**.

Step 7 Establish cross-Manager mutual trust between two clusters.

Step 8 Run the following commands to back up and modify the **workflow.xml** file:

cp workflow.xml workflow.xml.bak

vi workflow.xml

Modify the following content:

```
<workflow-app xmlns="uri:oozie:workflow:1.0" name="distcp-wf">
   <start to="distcp-node"/>
```

```
  <action name="distcp-node">
     <distcp xmlns="uri:oozie:distcp-action:1.0">
        <resource-manager>${resourceManager}</resource-manager>
        <name-node>${nameNode}</name-node>
        <prepare>
           <delete path="hdfs://target_ip:target_port/user/${userName}/${examplesRoot}/output-data/${outputDir}"/>
        </prepare>
        <configuration>
           <property>
              <name>mapred.job.queue.name</name>
              <value>${queueName}</value>
           </property>
           <property>
              <name>oozie.launcher.mapreduce.job.hdfs-servers</name>
              <value>hdfs://source_ip:source_port,hdfs://target_ip:target_port</value>
           </property>
        </configuration>
        <arg>${nameNode}/user/${userName}/${examplesRoot}/input-data/text/data.txt</arg>
        <arg>hdfs://target_ip:target_port/user/${userName}/${examplesRoot}/output-data/${outputDir}/data.txt</arg>
        </distcp>
     <ok to="end"/>
     <error to="fail"/>
  </action>
  <kill name="fail">
     <message>DistCP failed, error message[${wf:errorMessage(wf:lastErrorNode())}]</message>
  </kill>
  <end name="end"/>
</workflow-app>
```

target_ip:target_port is the HDFS active NameNode address of the other trusted cluster, for example, **10.10.10.233:25000**.

source_ip:source_port indicates the HDFS active NameNode address of the source cluster, for example, **10.10.10.223:25000**.

Change the two IP addresses and port numbers based on the site requirements.

Step 9 Run the **oozie job** command to run the workflow file:

oozie job -oozie https://*Host name of the Oozie role*:21003/oozie/ -config job.properties -run

NOTE

- The command parameters are described as follows:

  -oozie URL of the Oozie server that executes a job

  -config Workflow property file

  -run Executing a workflow

- If a job ID, for example, **job: 0000021-140222101051722-oozie-omm-W**, is displayed after the workflow file is executed, the job is successfully submitted. You can view the execution results on the Oozie management page.

  Log in to the Oozie web UI at **https**://*IP address of the Oozie role*:**21003/oozie** as user **oozieuser**.

  On the Oozie web UI, you can view the submitted workflow information based on the job ID in the table on the page.

----End

12.19.3.5 Submitting Other Jobs

Scenario

In addition to Hive, Spark2x, and Loader jobs, MapReduce, Java, Shell, HDFS, SSH, SubWorkflow, Streaming, and scheduled jobs can be submitted using the Oozie client.

NOTE

You are advised to download the latest client.

Prerequisites

- The Oozie component and its client have been installed and are running properly.
- You have created or obtained the human-machine account and password for accessing the Oozie service.

  NOTE

  - Shell job:

    This user must belong to the **hadoop** and **supergroup** groups and be assigned the Oozie role operation permission. The Shell script must have the execution permission on each NodeManager.
  - SSH job:

    This user must belong to the **hadoop** and **supergroup** groups and be assigned the Oozie role operation permission. The mutual trust configuration is complete.
  - Other jobs:

    This user must belong to the **hadoop** and **supergroup** groups and be assigned the Oozie role operation permission and other required permissions.
  - This user must also be assigned the **manager_viewer** role at least.
- You have obtained the URL of the Oozie server (any instance) in the running state, for example, **https://10.1.130.10:21003/oozie**.
- You have obtained the name of the Oozie server, for example, **10-1-130-10**.
- You have obtained the IP address of the active Yarn ResourceManager, for example, **10.1.130.11**.

Procedure

Step 1 Log in to the node where the Oozie client is installed as the client installation user.

Step 2 Run the following command to obtain the installation environment. **/opt/client/** is an example client installation path.

source /opt/client/bigdata_env

Step 3 Check the cluster authentication mode.

- If the cluster is in security mode, run the **kinit** command to authenticate users.

  For example, the **oozieuser** user is authenticated using the following command:

  kinit oozieuser

- If the cluster is in normal mode, go to **Step 4**.

Step 4 Go to the example directory based on the type of the task you submit.

Table 12-316 List of example directories

Job Type	Example Directory
MapReduce job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/map-reduce**
Java job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/java-main**
Shell job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/shell**
Streaming job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/shell**
SubWorkflow job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/subwf**
SSH job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/ssh**
Scheduled job	*Client installation directory*/**Oozie/oozie-client-*/examples/apps/cron**

NOTE

The examples of other jobs contain HDFS job examples.

Table 12-317 lists the files that you need to pay attention to in the example directory.

Table 12-317 File description

File	Description
job.properties	Parameter definition file of a workflow
workflow.xml	Rule definition file of a workflow
lib	Directory of the JAR file on which a workflow depends
coordinator.xml	Scheduled job configuration file which can be used to set a scheduled policy. The file is in the **cron** directory.
oozie_shell.sh	Shell script file required for submitting shell jobs. The file is in the **shell** directory.

Step 5 Run the following command to edit the **job.properties** file:

vi job.properties

Perform the following modifications:

Change the value of **userName** to the name of the human-machine user who submits the job, for example, **userName=oozieuser**.

Step 6 Run the **oozie job** command to run the workflow file:

oozie job -oozie https://*Host name of the oozie role***:21003/oozie -config** *File path of job.properties* **-run**

Example:

oozie job -oozie https://10-1-130-10:21003/oozie -config

/opt/client/Oozie/oozie-client-*/examples/apps/map-reduce/job.properties -run

NOTE

- The command parameters are described as follows:

  -oozie URL of the Oozie server that executes a job

  -config Workflow property file

  -run Executing a workflow

- If a job ID, for example, **job: 0000021-140222101051722-oozie-omm-W**, is displayed after the workflow file is executed, the job is successfully submitted. You can view the execution results on the Oozie management page.

  Log in to the Oozie web UI at **https**://*IP address of the Oozie role*:**21003/oozie** as user **oozieuser**.

  On the Oozie web UI, you can view the submitted workflow information based on the job ID in the table on the page.

----End

12.19.4 Using Hue to Submit an Oozie Job

12.19.4.1 Creating a Workflow

Scenario

You can submit an Oozie job on the Hue management page, but a workflow must be created before the job is submitted.

Prerequisites

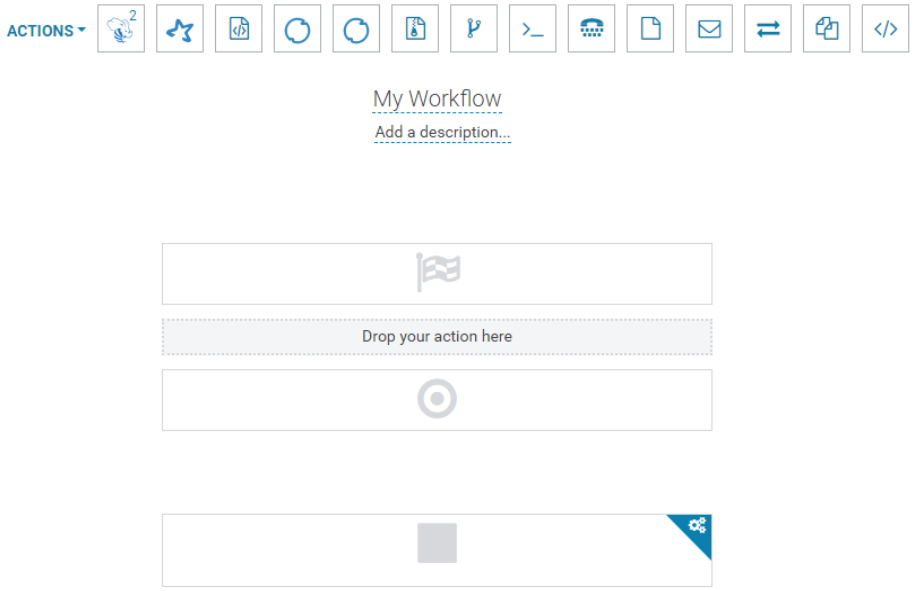
Before using Hue to submit an Oozie job, configure the Oozie client and upload the sample configuration file and JAR file to the specified HDFS directory. For details, see **Using the Oozie Client**.

Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the navigation tree on the left, click  and choose **Workflow** to open the Workflow editor.

Step 3 Select **Actions** from the **DOCUMENTS** drop-down list, select the job type to be created and drag it to the operation area.



For submitting different job types, follow instructions in the following sections:

- **Submitting a Hive2 Job**
- **Submitting a Spark2x Job**
- **Submitting a Java Job**
- **Submitting a Loader Job**
- **Submitting a MapReduce Job**
- **Submitting a Sub-workflow Job**
- **Submitting a Shell Job**
- **Submitting an HDFS Job**
- **Submitting a Streaming Job**
- **Submitting a DistCp Job**

----End

12.19.4.2 Submitting a Workflow Job

12.19.4.2.1 Submitting a Hive2 Job

Scenario

This section describes how to submit an Oozie job of the Hive2 type on the Hue web UI.

Procedure

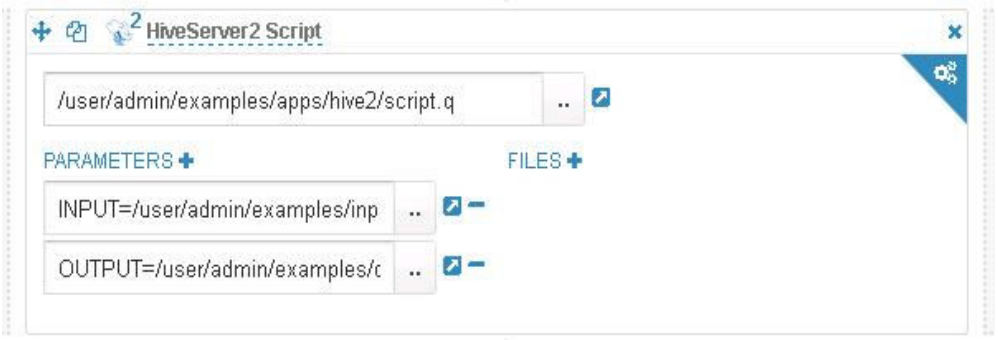
Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **HiveServer2 Script** and drag it to the operation area.

Step 3 In the **HiveServer2 Script** dialog box that is displayed, configure the script path in the HDFS, for example, **/user/admin/examples/apps/hive2/script.q**, and click **Add**.

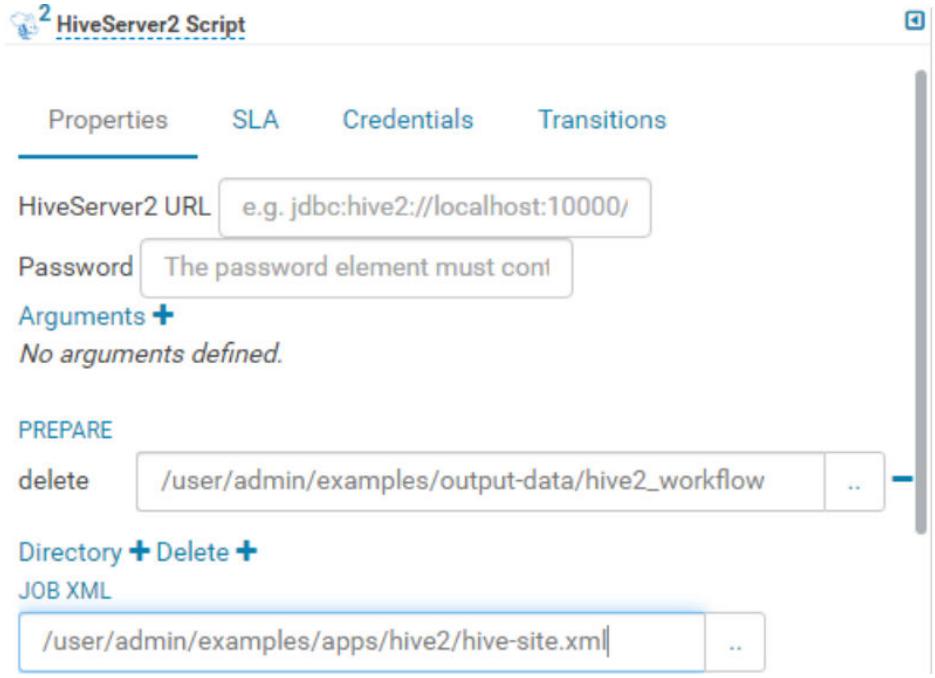
Step 4 Click **PARAMETER+** to add input and output parameters.

For example, if the input parameter is **INPUT=/user/admin/examples/input-data/table**, the output parameter is **OUTPUT=/user/admin/examples/output-data/hive2_workflow**.



Step 5 Click the configuration button  in the upper right corner. On the configuration page that is displayed, click **Delete +** to delete a directory, for example, **/user/admin/examples/output-data/hive2_workflow**.

Step 6 Configure the job XML, for example, to the HDFS path **/user/admin/examples/apps/hive2/hive-site.xml**.



NOTE

If the preceding parameters and values are modified, you can query them in **Oozie client installation directory//oozie-client-*/conf/hive-site.xml**.

Step 7 Click in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Hive2-Workflow**.

Step 8 After the configuration is saved, click , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.2 Submitting a Spark2x Job

Scenario

This section describes how to submit an Oozie job of the Spark2x type on Hue.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select next to **Spark program** and drag it to the operation area.

Step 3 In the Spark window that is displayed, set the value of **Files**, for example, to **hdfs://hacluster/user/admin/examples/apps/spark2x/lib/oozie-examples.jar**.

Set the value of **jar/py name**, for example, to **org.apache.oozie.example.SparkFileCopy**, and click **Add**.

Step 4 Set the value of **Main class**, for example, **org.apache.oozie.example.SparkFileCopy**.

Step 5 Click **PARAMETER+** to add related input and output parameters.

For example, add the following parameters:

- **hdfs://hacluster/user/admin/examples/input-data/text/data.txt**
- **hdfs://hacluster/user/admin/examples/output-data/spark_workflow**

Step 6 In the **Options list** text box, specify Spark parameters, for example, **--conf spark.yarn.archive=hdfs://hacluster/user/spark2x/jars/8.1.0.1/spark-archive-2x.zip --conf spark.eventLog.enabled=true --conf spark.eventLog.dir=hdfs://hacluster/spark2xJobHistory2x**.

NOTE

The version 8.1.0.1 is used as an example. Replace it with the actual version number.

Step 7 Click the configuration button  in the upper right corner. Set the value of **Spark Master**, for example, to **yarn-cluster**. Set the value of **Mode**, for example, **cluster**.

Step 8 On the configuration page that is displayed, click **Delete +** to delete a directory, for example, **hdfs://hacluster/user/admin/examples/output-data/spark_workflow**.

Step 9 Click **PROPERTIES+** and add **sharelib** used by Oozie. Enter the attribute name **oozie.action.sharelib.for.spark** in the left text box and the attribute value **spark2x** in the right text box.

Step 10 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Spark-Workflow**.

Step 11 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.3 Submitting a Java Job

Scenario

This section describes how to submit an Oozie job of the Java type on the Hue web UI.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **Java program** and drag it to the operation area.

Step 3 In the **Jar program** window that is displayed, set the value of **Jar name**, for example, **/user/admin/examples/apps/java-main/lib/oozie-examples-5.1.0.jar**. Set the value of **Main class**, for example, **org.apache.oozie.example.DemoJavaMain**. Click **Add**.

Step 4 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Java-Workflow**.

Step 5 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.4 Submitting a Loader Job

Scenario

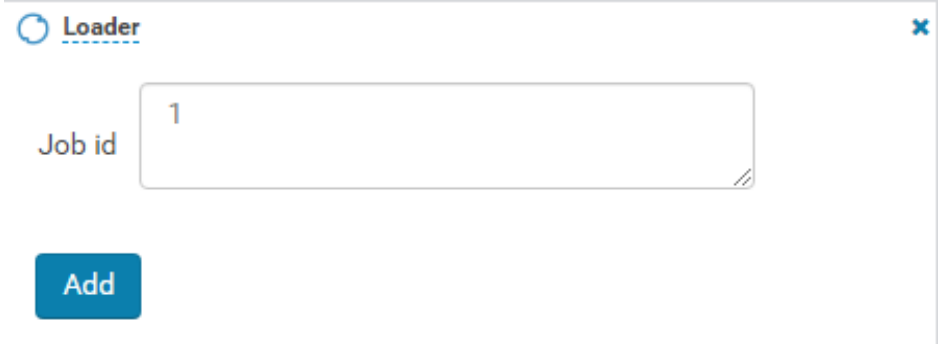
This section describes how to submit an Oozie job of the Loader type on the Hue web UI.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **Loader** and drag it to the operation area.

Step 3 In the **Loader** window that is displayed, set **Job id**, for example, to **1**. Click **Add**.



NOTE

Job id is the ID of the Loader job to be orchestrated and can be obtained from the Loader page.

You can create a Loader job to be scheduled and obtain its job ID. For details, see **Using Loader**.

Step 4 Click in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Loader-Workflow**.

Step 5 After the configuration is saved, click , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.5 Submitting a MapReduce Job

Scenario

This section describes how to submit an Oozie job of the MapReduce type on the Hue web UI.

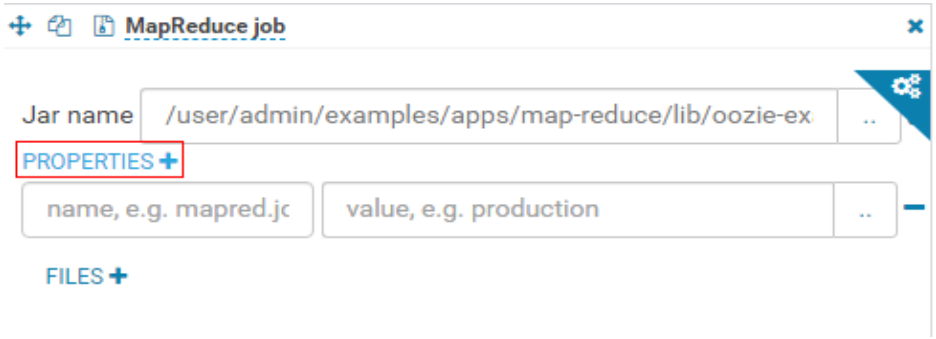
Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select next to **MapReduce job** and drag it to the operation area.

Step 3 In the displayed **MapReduce job** dialog box, set **Jar name**, for example, to **/user/admin/examples/apps/map-reduce/lib/oozie-examples-5.1.0.jar**. Click **Add**.

Step 4 Click **PROPERTIES+** to add input and output properties.



For example, set the value of **mapred.input.dir** to **/user/admin/examples/input-data/text** and set the value of **mapred.output.dir** to **/user/admin/examples/output-data/map-reduce_workflow**.

Step 5 Click the configuration button  in the upper right corner. On the configuration page that is displayed, click **Delete +** to delete a directory, for example, **/user/admin/examples/output-data/map-reduce_workflow**.

Step 6 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **MapReduce-Workflow**.

Step 7 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.6 Submitting a Sub-workflow Job

Scenario

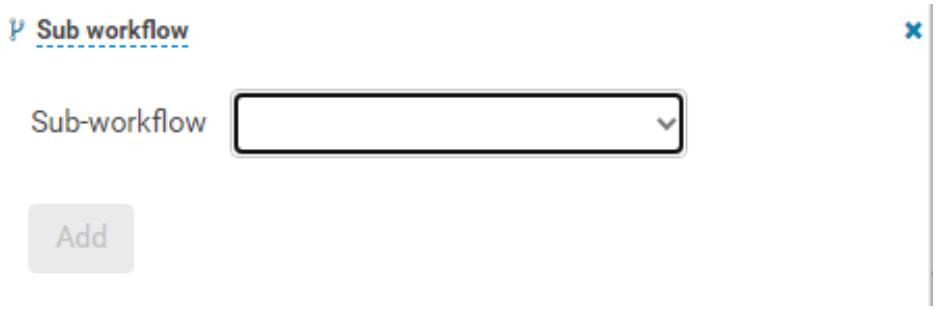
This section describes how to submit an Oozie job of the Sub-workflow type on the Hue web UI.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **Sub workflow** and drag it to the operation area.

Step 3 In the **Sub workflow** dialog box that is displayed, set **Sub-workflow**, for example, to **Java-Workflow** (one of the created workflows) from the drop-down list box, and click **Add**.



Step 4 Click in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Subworkflow-Workflow**.

Step 5 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.7 Submitting a Shell Job

Scenario

This section describes how to submit an Oozie job of the Shell type on the Hue web UI.

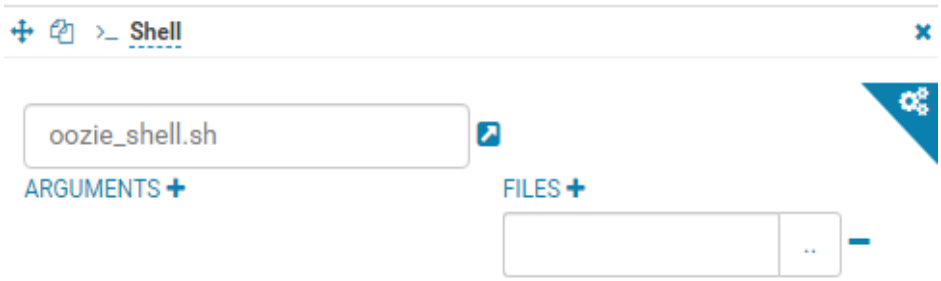
Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **Shell** and drag it to the operation area.

Step 3 In the **Shell** window that is displayed, set **Shell command**, for example, to **oozie_shell.sh**, and click **Add**.

Step 4 Click **FILE+** to add the Shell command execution file and Oozie example execution file, for example, **/user/admin/examples/apps/shell/oozie_shell.sh**.



Step 5 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Shell-Workflow**.

Step 6 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

NOTE

- When configuring a shell command as a Linux command, specify it as the original command instead of the shortcut key command. For example, do not set **ls -l** to **ll**. You can configure it as the shell command **ls**, and add a parameter **-l**.
- When uploading the shell script to HDFS on Windows, make sure that the shell script format is Unix. If the format is incorrect, the shell job fails to be submitted.

----End

12.19.4.2.8 Submitting an HDFS Job

Scenario

This section describes how to submit an Oozie job of the HDFS type on the Hue web UI.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **Fs** and drag it to the operation area.

Step 3 In the **Fs** window that is displayed, click **Add**.

Step 4 Click **CREATE DIRECTORY+** to add the HDFS directories to be created, for example, **/user/admin/examples/output-data/mkdir_workflow** and **/user/admin/examples/output-data/mkdir_workflow1**.

Step 5 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **HDFS-Workflow**.

Step 6 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.9 Submitting a Streaming Job

Scenario

This section describes how to submit an Oozie job of the Streaming type on the Hue web UI.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select  next to **Streaming** and drag it to the operation area.

Step 3 In the **Streaming** window that is displayed, set **Mapper**, for example, to **/bin/cat**. Set **Reducer**, for example, to **/usr/bin/wc**. Click **Add**.

Step 4 Click **FILE+** to add the files required for running.

for example, **/user/oozie/share/lib/mapreduce-streaming/hadoop-streaming-3.1.1.jar** and **/user/oozie/share/lib/mapreduce-streaming/oozie-sharelib-streaming-5.1.0.jar**.

Step 5 Click the configuration button  in the upper right corner. On the configuration page that is displayed, click **Delete+** to delete a directory, for example, **/user/admin/examples/output-data/streaming_workflow**.

Step 6 Click **PROPERTIES+** to add the following properties:

- Enter the property name **mapred.input.dir** in the left box and enter the property value **/user/admin/examples/input-data/text** in the right box.
- Enter the property name **mapred.output.dir** in the left box and enter the attribute value **/user/admin/examples/output-data/streaming_workflow** in the right box.

Step 7 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Streaming-Workflow**.

Step 8 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.10 Submitting a DistCp Job

Scenario

This section describes how to submit an Oozie job of the DistCp type on the Hue web UI.

Procedure

Step 1 Create a workflow. For details, see **Creating a Workflow**.

Step 2 On the workflow editing page, select next to **Distcp** and drag it to the operation area.

Step 3 Determine whether the current DistCp operation is performed across clusters.

- If yes, go to **Step 4**.
- If no, go to **Step 7**.

Step 4 Establish cross-Manager mutual trust between two clusters.

Step 5 In the **Distcp** window that is displayed, set the value of **Source**, for example, to **hdfs://hacluster/user/admin/examples/input-data/text/data.txt**. Set **Destination**, for example, to **hdfs://target_ip:target_port/user/admin/examples/output-data/distcp-workflow/data.txt**. Click **Add**.

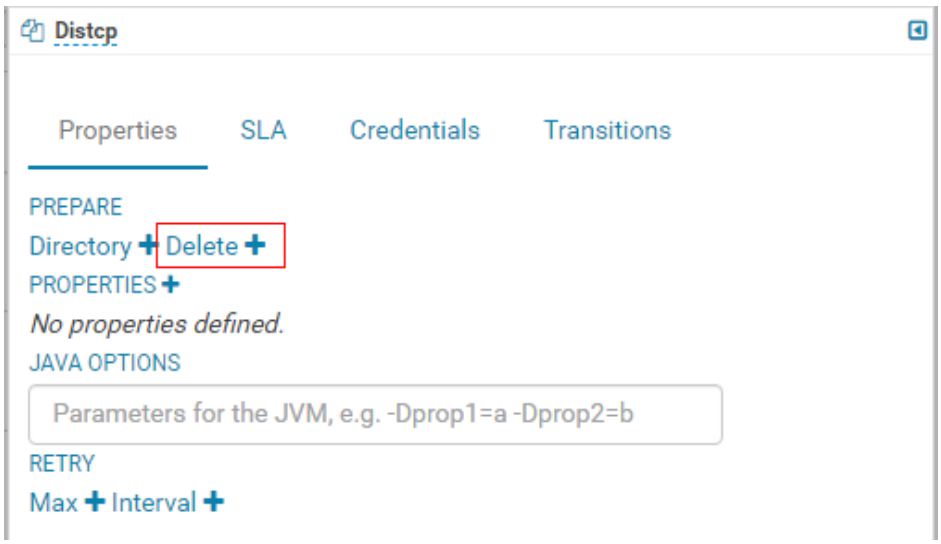
Step 6 Click the configuration button in the upper right corner. On the **Properties** tab page, click **PROPERTIES+**, enter the attribute name **oozie.launcher.mapreduce.job.hdfs-servers** in the text box on the left, enter the attribute value **hdfs://***source_ip:source_port*,**hdfs://***target_ip:target_port* in the text box on the right, and go to **Step 8**.

NOTE

source_ip: service address of the HDFS NameNode in the source cluster
source_port: port number of the HDFS NameNode in the source cluster.
target_ip: service address of the HDFS NameNode in the target cluster
target_port: port number of the HDFS NameNode in the target cluster.

Step 7 In the **Distcp** window that is displayed, set the value of **Source**, for example, to **/user/admin/examples/input-data/text/data.txt**. Set **Destination**, for example, to **/user/admin/examples/output-data/distcp-workflow/data.txt**. Click **Add**.

Step 8 Click in the upper right corner. On the configuration page that is displayed, click **Delete+** and add the directory to be deleted, for example, **/user/admin/examples/output-data/distcp-workflow**.



Step 9 Click in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Distcp-Workflow**.

Step 10 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.11 Example of Mutual Trust Operations

Scenario

This section guides you to enable unidirectional password-free mutual trust when Oozie nodes are used to execute shell scripts of external nodes through SSH jobs.

Prerequisites

You have installed Oozie, and it can communicate with external nodes (nodes connected using SSH).

Procedure

Step 1 Ensure that the user used for SSH connection exists on the external node, and the user directory **~/.ssh** exists.

Step 2 Log in to the Oozie node as user **omm** and run the **ssh-keygen -t rsa** command to generate public and private keys.

Step 3 Run the **cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys** statement to add the public key to the **authorized_keys** file.

Step 4 Upload the **id_rsa.pub** file to an existing directory, for example, **/opt/**, on the external node as user **root**.

scp ~/.ssh/id_rsa.pub root@*IP address of the external node***:/opt/id_rsa.pub**

Step 5 Log in to the external node where the shell is located and go to the directory described in **Step 4**. The **id_rsa.pub** file can be found.

Run the **cat id_rsa.pub >> ~/.ssh/authorized_keys** statement to add the public key to the **authorized_keys** file of the shell user.

Step 6 Change the permission on the directory.

chmod 700 ~/.ssh

chmod 600 /opt/id_rsa.pub

chmod 600 ~/.ssh/authorized_keys

NOTE

- The user of the node where shell resides (external node) has the permission to execute shell scripts and access all directories and files involved in the Shell scripts.
- If Oozie has multiple nodes, perform **Step 2** to **Step 6** on all Oozie nodes.

----End

12.19.4.2.12 Submitting an SSH Job

Scenario

This section guides you to submit an Oozie job of the SSH type on the Hue web UI.

Due to security risks, SSH jobs cannot be submitted by default. To use the SSH function, you need to manually enable it.

Procedure

Step 1 Enable the SSH function.

1. On FusionInsight Manager, choose **Cluster** > **Services** > **Oozie** and click the **Configurations** tab and then **All Configurations**. In the navigation pane on the left, choose **oozie(Role)** > **Security**, change the value of **oozie.job.ssh.enable** to **true**, and click **Save**. In the displayed dialog box, click **OK** to save the configuration.
2. On the **Dashboard** page of Oozie, choose **More** > **Restart Service** in the upper-right corner to restart Oozie.

Step 2 Create a workflow. For details, see **Creating a Workflow**.

Step 3 For details about how to add the trust relationship, see **Example of Mutual Trust Operations**.

Step 4 On the workflow editing page, select the **Ssh** button  and drag it to the operation area.

Step 5 In the **Ssh** window that is displayed, set **User and Host** and **Ssh command** commands and click **Add**.

Step 6 Click  in the upper right corner of the Oozie editor.

If you need to modify the job name before saving the job (default value: **My Workflow**), click the name directly for modification, for example, **Ssh-Workflow**.

Step 7 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.2.13 Submitting a Hive Script

Scenario

This section describes how to submit a Hive job on the Hue web UI.

Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the navigation tree on the left, click  and choose **Workflow** to open the Workflow editor.

Step 3 Click **Documents**, click  to select a Hive script from the operation list, and drag it to the operation page.

Step 4 In the **HiveServer2 Script** dialog box that is displayed, select the saved Hive script. For details about how to save the Hive script, see **Using HiveQL Editor on the Hue Web UI**. Select a script and click **Add**.

Step 5 Configure the Job XML, for example, to the HDFS path **/user/admin/examples/apps/hive2/hive-site.xml**. For details, see **Submitting a Hive2 Job**.

Step 6 Click  in the upper right corner of the Oozie editor.

Step 7 After the configuration is saved, click  , and submit the job.

After the job is submitted, you can view the related contents of the job, such as the detailed information, logs, and processes, on Hue.

----End

12.19.4.3 Submitting a Coordinator Periodic Scheduling Job

Scenario

This section describes how to submit a job of the periodic scheduling type on the Hue web UI.

Prerequisites

Required workflow jobs have been configured before the coordinator task is submitted.

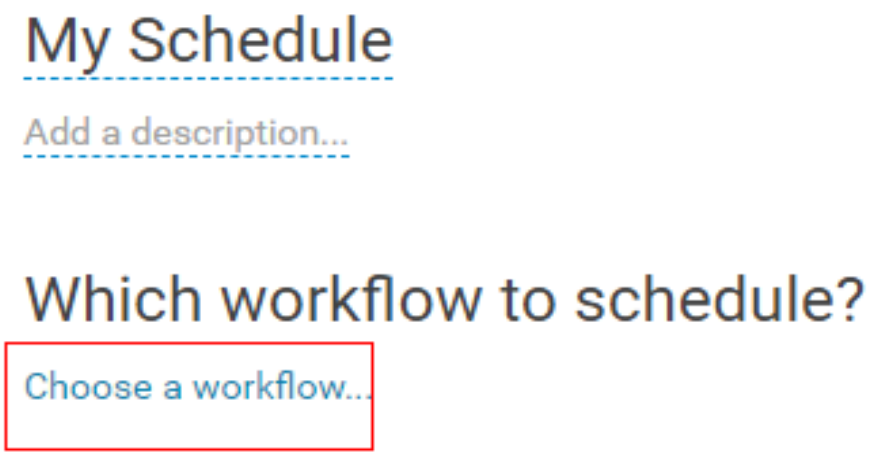
Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 In the navigation tree on the left, click  and choose **Schedule** to open the Coordinator editor.

Step 3 On the job editing page, click **My Schedule** to change the job name.

Step 4 Click **Choose a Workflow...** to select the workflow to be orchestrated.



Step 5 Select a workflow, set the job execution frequency as prompted, and click  in the upper right corner to save the workflow job.

NOTE

Because the time zone is changed, the difference between the time and the local time may be several hours.

Step 6 Click  in the upper right corner of the editor, set the start value and end value of the time range for executing the scheduled job, and click **Submit** to submit the job.

NOTE

Because the time zone is changed, the difference between the time and the local time may be several hours.

----End

12.19.4.4 Submitting a Bundle Batch Processing Job

Scenario

In the case that multiple scheduled jobs exist at the same time, you can manage the jobs in batches over the Bundle task. This section describes how to submit a job of the batch type on the Hue web UI.

Prerequisites

Required related workflow and Coordinator jobs have been configured before the Bundle batch processing job is submitted.

Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

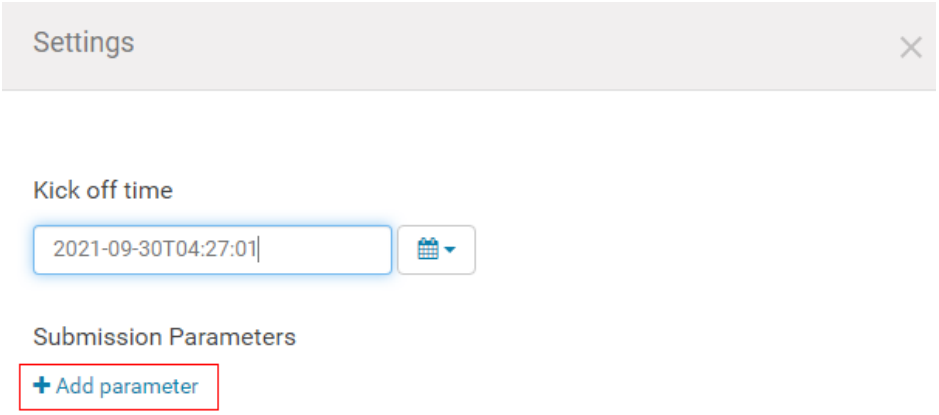
Step 2 In the navigation tree on the left, click  and choose **Bundle** to open the Bundle editor.

Step 3 On the job editing page, click **My Bundle** to change the job name.

Step 4 Click **+Add a coordinator** to select the Coordinator job to be orchestrated.

Step 5 Set the start time and the end time for the scheduled coordinator jobs as prompted and click  in the upper right corner to save the job.

Step 6 Click in the upper right corner of the editor, select  from the displayed menu, set the start time of the bundle task, click **+Add parameter** to add parameters, and close the dialog box to save the settings.



NOTE

Because the time zone is changed, the difference between the time and the local time may be several hours.

Step 7 Click  in the upper right corner of the editor. In the dialog box that is displayed, click **Submit** to submit the job.

----End

12.19.4.5 Querying the Operation Results

Scenario

After the jobs are submitted, you can view the execution status of a specific job on Hue.

Procedure

Step 1 Access the Hue web UI. For details, see **Accessing the Hue Web UI**.

Step 2 Click  . On the displayed page, you can view information about the Workflow, Schedule, and Bundle tasks.

----End

12.19.5 Oozie Log Overview

Log Description

Log path: The default storage paths of Oozie log files are as follows:

- Run log: **/var/log/Bigdata/oozie**
- Audit log: **/var/log/Bigdata/audit/oozie**

Log archiving rule: Oozie logs are classified into run logs, script logs, and audit logs. The maximum size of a run log file is 20 MB, and a maximum of 20 run log files can be reserved. The maximum size of an audit log file is 20 MB, and a maximum of 20 audit log files can be reserved.

NOTE

A compressed log file is generated for **oozie.log** every hour. 720 compressed files (log files of one month) are retained by default.

Table 12-318 Oozie log list

Log Type	Log File Name	Description
Run log	jetty.log	Oozie built-in jetty server log file, which is used to process the request and response information of OozieServlet
	jetty.out	Oozie process startup log file
	oozie_db_temp.log	Oozie database connection log
	oozie-instrumentation.log	Oozie dashboard log file, which records the Oozie running status and configuration information of each component
	oozie-jpa.log	openJPa run log file
	oozie.log	Oozie run log file
	oozie-<SSH_USER>-<DATE>-<PID>-gc.log	Log file that records the garbage collection of the Oozie service
	oozie-ops.log	Oozie operation log file
	check-serviceDetail.log	Oozie health check logs

Log Type	Log File Name	Description
	oozie-error.log	Oozie running error logs
	threadDump-<DATE>.log	Log file that records stack information when the service process exits normally
Script logs	postinstallDetail.log	Work log file generated after the installation and before the startup
	prestartDetail.log	Pre-startup log file
	startDetail.log	Service startup log file
	stopDetail.log	Service stop log file
	upload-sharelib.log	Operation logs uploaded by **sharelib**
Audit log	oozie-audit.log	Audit log

Log Level

Table 12-319 describes the log levels provided by Oozie.

The priorities of log levels are ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the set level are printed. The number of printed logs decreases as the configured log level increases.

Table 12-319 Log levels

Level	Description
ERROR	Logs of this level record abnormal information about events that cause process exceptions.
WARN	Logs of this level record exception information about the current event processing.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record system information and information about database underlying data transmission.

To modify log levels, perform the following operations:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Oozie** > **Configurations**.

Step 3 Select **All Configurations**.

Step 4 On the menu bar on the left, select the log menu of the target role.

Step 5 Select a desired log level.

Step 6 Click **Save**, and then click **OK**. The settings take effect after the processing is complete.

----End

Log Formats

The following table lists the Oozie log formats.

Table 12-320 Log formats

Log Type	Format	Example
Run log	*<yyyy-MM-dd HH:mm:ss,SSS><Log level><Location where the log event occurs><Log level><Message in the log>*	2015-05-29 21:01:45,268 INFO StatusTransitService$StatusTransitRunnable:539 - USER[-] GROUP[-] Released lock for [org.apache.oozie.service.StatusTransitService]
Script logs	*<yyyy-MM-dd HH:mm:ss,SSS><Host name > <Log level > <Message in the log>*	2015-06-01 17:18:03 001 suse11-192-168-0-111 oozie INFO Running oozie service check script
Audit log	*<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|< Thread name > | Message in the log | Location where the log event occurs*	2015-06-01 22:38:41,323 | INFO | http-bio-21003-exec-8 | IP [192.168.0.111] USER [null], GROUP [null], APP [null], JOBID [null], OPERATION [null], PARAMETER [null], RESULT [SUCCESS], HTTPCODE [200], ERRORCODE [null], ERRORMESSAGE [null] | org.apache.oozie.util.XLog.log(XLog.java:539)

12.19.6 Common Issues About Oozie

12.19.6.1 Oozie Scheduled Tasks Are Not Executed on Time

Question

Why are not Coordinator scheduled jobs executed on time on the Hue or Oozie client?

Answer

Use UTC time. For example, set **start=2016-12-20T09:00Z** in **job.properties** file.

12.19.6.2 Why Update of the share lib Directory of Oozie on HDFS Does Not Take Effect?

Symptom

A new JAR package is uploaded to the **/user/oozie/share/lib** directory on HDFS. However, an error indicating that the class cannot be found is reported during task execution.

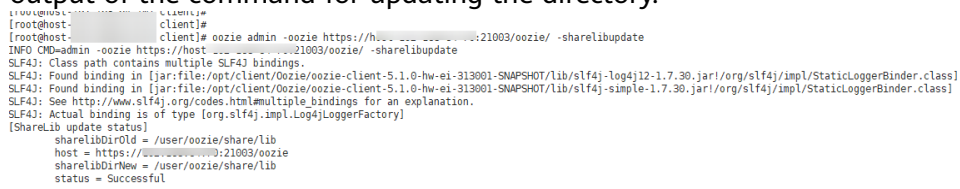
Solution

Run the following command on the client to refresh the directory:

oozie admin -oozie https://xxx.xxx.xxx.xxx:21003/oozie -sharelibupdate

12.19.6.3 Common Oozie Troubleshooting Methods

1. Check the job logs on Yarn. Run the command executed through Hive SQL using beeline to ensure that Hive is running properly.
2. If error information such as "classnotfoundException" is displayed, check whether the JAR package of the faulty class exists in the **/user/oozie/share/lib** directory of each component. If no, add the JAR package and go to **Why Update of the share lib Directory of Oozie on HDFS Does Not Take Effect?**. If the faulty class still cannot be found after the **share lib** directory is updated, check whether **sharelibDirNew** is **/user/oozie/share/lib** in the output of the command for updating the directory.



3. If "NosuchMethodError" is displayed, check whether the JAR packages of each component in the **/user/oozie/share/lib** directory have multiple versions. Note that the JAR packages uploaded by the service cannot conflict with each other. You can check whether a JAR package conflict occurs based on the loaded JAR packages in Oozie run logs on Yarn.
4. If the self-developed code is abnormal, run the Oozie sample to check whether Oozie is running properly.
5. Contact technical support personnel. By using this method, you must collect run logs of Oozie on Yarn, Oozie logs, and component run logs. For example, if an exception occurs when Hive runs on Oozie, you need to collect Hive logs.

12.20 Using Presto

12.20.1 Accessing the Presto Web UI

You can view the Presto statistics on the graphical Presto web UI. You are advised to use Google Chrome to access the Presto web UI because it cannot be accessed using Internet Explorer.

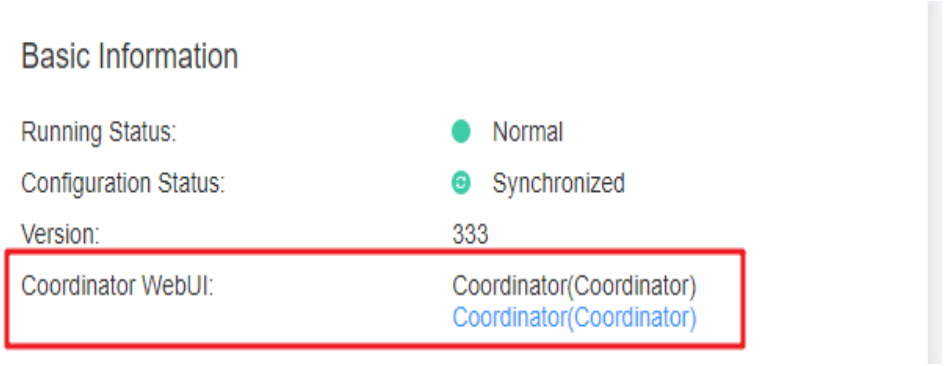
Prerequisites

- Presto has been installed in a cluster.
- The cluster client has been installed, for example, in the **/opt/client** directory. The client directory in the following operations is only an example. Change it based on the actual installation directory onsite.

Accessing the Presto Web UI

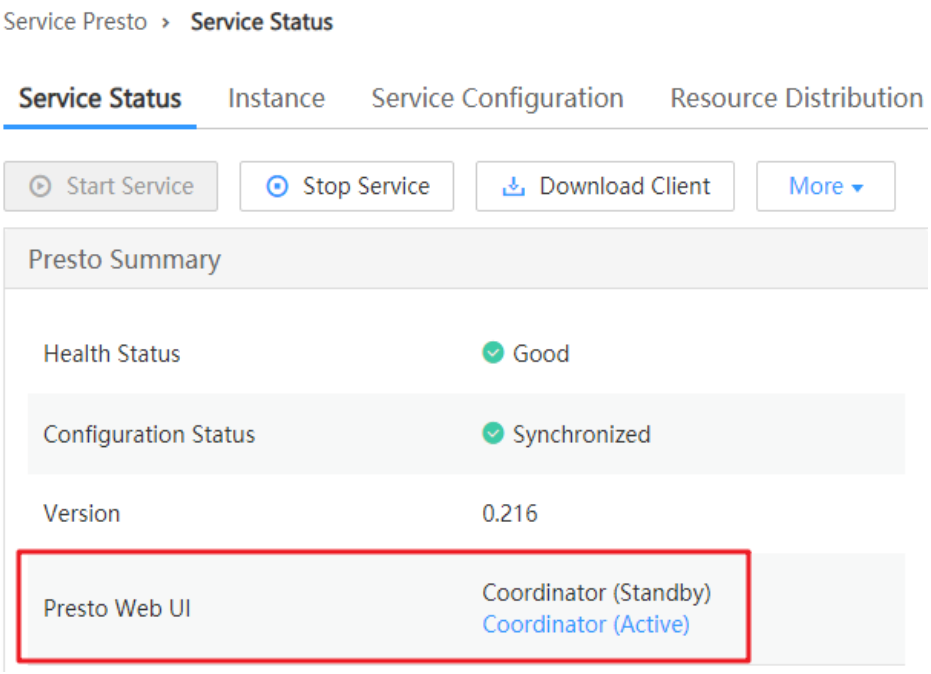
- Method 1 (for MRS 3.*x* or later)
  a. Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the target cluster* > **Services**.
  b. Select **Presto**. In the **Basic Information** area, click **Coordinator(Coordinator)** next to **Coordinator WebUI**. The Coordinator web UI is displayed.

  Figure 12-39 Coordinator WebUI

- Method 2 (for versions earlier than MRS 3.*x*)
  a. Log in to MRS Manager and choose **Services**.
  b. Select **Presto**. In the **Presto Summary** area, click **Coordinator (Active)** next to **Presto Web UI**. The Presto web UI is displayed.

Figure 12-40 Presto WebUI



NOTE

When accessing the Presto web UI for the first time, you must add the address to the trusted site list.

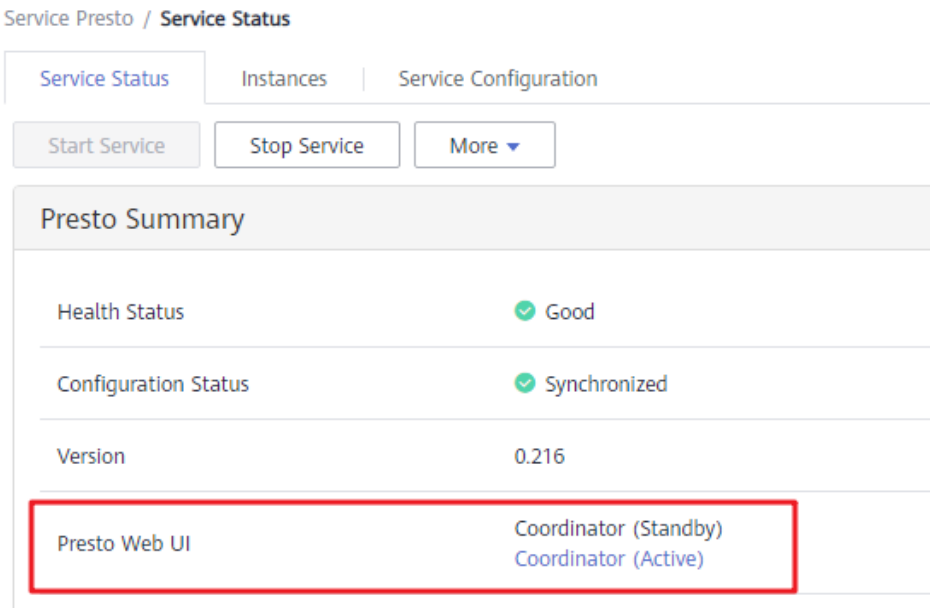
- Method 3 (for MRS 1.9.2 or later)
  a. Log in to the MRS console, click the target cluster name to go to the cluster details page, and click the **Components** tab.

     NOTE

     If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

  b. Click **Presto**. In the **Presto Summary** area, click **Coordinator (Active)** next to **Presto Web UI**. The Presto web UI is displayed.

Figure 12-41 Presto WebUI



12.20.2 Using a Client to Execute Query Statements

You can perform an interactive query on an MRS cluster client. For clusters with Kerberos authentication enabled, users who submit topologies must belong to the **presto** group.

The Presto component of MRS 3.x does not support Kerberos authentication.

Prerequisites

- The password of user **admin** has been obtained. The password of user **admin** is specified by the user during MRS cluster creation.
- The client has been updated.
- The Presto client has been manually installed for MRS 3.*x* clusters.

Procedure

Step 1 For clusters with Kerberos authentication enabled, log in to MRS Manager and create a role with the **Hive Admin Privilege** permission.

Step 2 Create a user that belongs to the **Presto** and **Hive** groups, bind the role created in **Step 1** to the user, and download the user authentication file.

Step 3 Upload the downloaded **user.keytab** and **krb5.conf** files to the node where the MRS client resides.

NOTE

For clusters with Kerberos authentication enabled, **Step 2** to **Step 3** must be performed. For normal clusters, start from **Step 4**.

Step 4 Prepare a client based on service conditions and log in to the node where the client is installed.

Step 5 Run the following command to switch the user:

sudo su - omm

Step 6 Run the following command to switch to the client directory, for example, **/opt/client**.

cd /opt/client

Step 7 Run the following command to configure environment variables:

source bigdata_env

Step 8 Connect to the Presto Server. The following provides two client connection methods based on the client type.

- Using the client provided by MRS
  - For clusters with Kerberos authentication disabled, run the following command to connect to the Presto Server of the cluster:

    presto_cli.sh
  - For clusters with Kerberos authentication disabled, run the following command to connect to the Presto Server of other clusters. In the command, **ip** indicates the floating IP address of the cluster Presto Server, which can be obtained by searching for **PRESTO_COORDINATOR_FLOAT_IP** in the Presto configuration items. **port** indicates the Presto Server port number and is set to **7520** by default.

    presto_cli.sh --server *http://ip:port*
  - For clusters with Kerberos authentication enabled, run the following command to connect to the Presto Server of the cluster:

    presto_cli.sh --krb5-config-path *krb5.conf file path* --krb5-principal *User's principal* --krb5-keytab-path *user.keytab file path* --user *presto username*
  - For clusters with Kerberos authentication enabled, run the following command to connect to the Presto Server of other clusters. In the command, **ip** indicates the floating IP address of the cluster Presto Server, which can be obtained by searching for **PRESTO_COORDINATOR_FLOAT_IP** in the Presto configuration items. **port** indicates the Presto Server port number and is set to **7521** by default.

    presto_cli.sh --krb5-config-path *krb5.conf file path* --krb5-principal *User's principal* --krb5-keytab-path *user.keytab file path* --server *https://ip:port* --krb5-remote-service-name *Presto Server name*
- Using the native client

  The native client of Presto is **Presto/presto/bin/presto** in the client directory.

Step 9 Run a query statement, for example, **show catalogs**.

NOTE

For clusters with Kerberos authentication enabled, when querying **Hive Catalog** data, the user who runs the Presto client must have the permission to access Hive tables and run the **grant all on table [table_name] to group hive** command in Hive beeline to grant permissions to the Hive group.

Step 10 After the query is complete, run the following command to exit the client:

quit

----End

12.21 Using Ranger (MRS 3.*x*)

12.21.1 Logging In to the Ranger Web UI

Ranger provides a centralized permission management framework to implement fine-grained permission control on components such as HDFS, HBase, Hive, and Yarn. In addition, Ranger also provides a web UI for administrators to perform operations.

Ranger User Type

Ranger users are classified into **admin**, **user**, and **auditor**. Different users have different permissions to view and operate the Ranger management interface.

- **Admin**: A security administrator can view all page content, manage permission management plug-ins and access control policies, view audit information, and set user types.
- **Auditor**: An audit administrator can view the permission management plug-ins and access control policies.
- **User**: A common user who can be assigned with specific permissions by the administrator.

Logging In to the Ranger Web UI

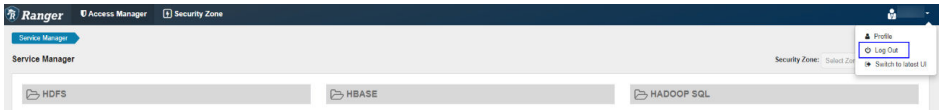
Security mode (Kerberos authentication is enabled for clusters)

Step 1 Log in to FusionInsight Manager as user **admin**. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > **Services** > **Ranger**. The Ranger service overview page is displayed.

Step 2 Click **RangerAdmin** in the **Basic Information** area. The Ranger web UI is displayed.

- The **admin** user in Ranger belongs to the **User** type and can only view the **Access Manager** as well as **Security Zone** pages.
- To view all management pages, switch to user **rangeradmin** or other users who have the Ranger administrator permissions.

  a. On the Ranger WebUI, click the user name in the upper right corner and choose **Log Out** to log out of the Ranger WebUI.



b. Log in to the system as user **rangeradmin** (default password: **Rangeradmin@123**) or another user who has the Ranger administrator permissions.

----End

Normal mode (Kerberos authentication is disabled for clusters)

Step 1 Log in to FusionInsight Manager as user **admin**. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > **Services** > **Ranger**. The Ranger service overview page is displayed.

Step 2 Click **RangerAdmin** in the **Basic Information** area. The Ranger web UI is displayed.

The **admin** user in Ranger belongs to the **Admin** type and can view all management pages of Ranger without switching to user **rangeradmin**.

NOTE

When a user logs in to the Ranger WebUI as user **rangeradmin** in normal mode, error 401 is reported.

----End

On the homepage of Ranger web UI, you can view the permission management plug-ins of the services integrated in Ranger. The plug-ins can be used to set more fine-grained permissions. For details about functions of main operations you can perform on the page, see **Table 12-321**.

Table 12-321 Functions of each operation portal on the Ranger page

Portal	Function
Access Manager	You can view the permission management plug-ins of each service integrated in Ranger. The plug-ins can be used to set more fine-grained permissions. For details, see **Configuring Component Permission Policies**.
Audit	You can view the audit logs related to Ranger running and permission control. For details, see **Viewing Ranger Audit Information**.
Security Zone	Administrators can divide resources of each component into multiple security zones where different administrators set security policies for specified resources of services to facilitate management. For details, see **Configuring a Security Zone**.
Settings	You can view Ranger permission settings, such as users, user groups, and roles. For details, see **Viewing Ranger Permission Information**.

12.21.2 Enabling Ranger Authentication

Scenario

This section guides you how to enable Ranger authentication. Ranger authentication is enabled by default in security mode and disabled by default in normal mode.

Procedure

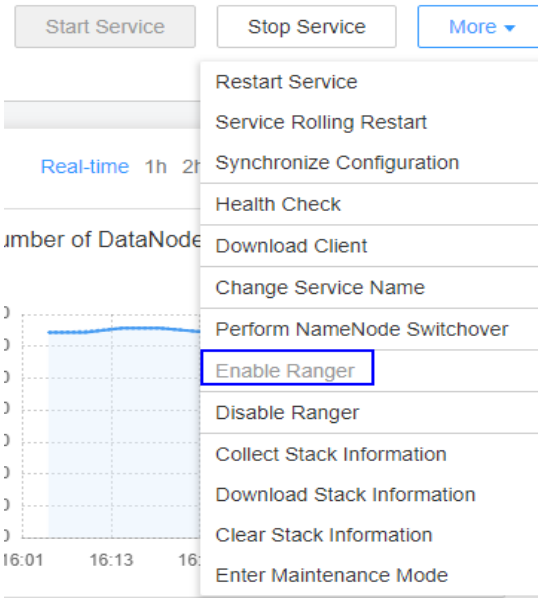
Step 1 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > **Services** > *Name of the service for which Ranger authentication is enabled*.

Step 2 In the upper right corner of the **Dashboard** page, click **More** and select **Enable Ranger**. In the displayed dialog box, enter the password and click **OK**. After the operation is successful, click **Finish**.

NOTE

If **Enable Ranger** is dimmed, Ranger authentication is enabled, as shown in **Figure 12-42**.

Figure 12-42 Enabling Ranger Authentication



Step 3 Perform a rolling service restart or restart the service.

----End

12.21.3 Configuring Component Permission Policies

In the newly installed MRS cluster, Ranger is installed by default, with the Ranger authentication model enabled. The administrator can set fine-grained security policies for accessing component resources through the component permission plug-ins.

Currently, the following components in a cluster in security mode support Ranger: HDFS, Yarn, HBase, Hive, Spark2x, Kafka, Storm..

Configuring User Permission Policies Using Ranger

Step 1 Log in to the Ranger management page as the administrator.

Step 2 In the **Service Manager** area on the Ranger homepage, click the permission plug-in name of a component. The page for security access policy list of the component is displayed.

NOTE

In the policy list of each component, many items are generated by default to ensure the permissions of some default users or user groups (such as the **supergroup** user group). Do not delete these items. Otherwise, the permissions of the default users or user groups are affected.

Step 3 Click **Add New Policy** and configure resource access policies for related users or user groups based on the service scenario plan.

The following policies are examples for different components:

- **Adding a Ranger Access Permission Policy for HDFS**
- **Adding a Ranger Access Permission Policy for HBase**
- **Adding a Ranger Access Permission Policy for Hive**
- **Adding a Ranger Access Permission Policy for Yarn**
- **Adding a Ranger Access Permission Policy for Spark2x**
- **Adding a Ranger Access Permission Policy for Kafka**
- **Adding a Ranger Access Permission Policy for Storm**

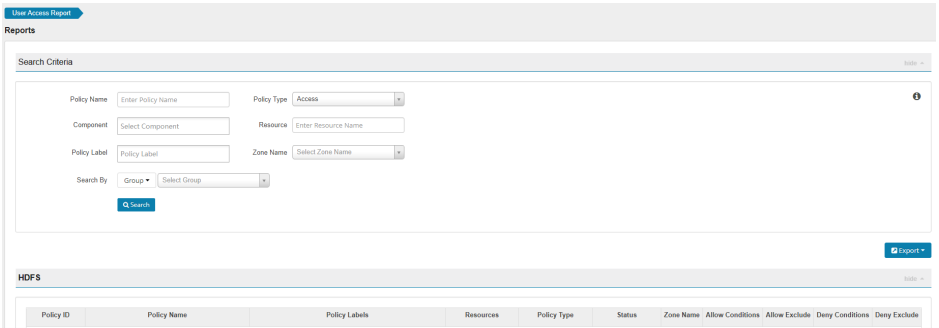
After the policies are added, wait for about 30 seconds for them to take effect.

NOTE

Each time a component is started, the system checks whether the default Ranger service of the component exists. If the service does not exist, the system creates the Ranger service and adds a default policy for it. If a service is deleted by mistake, you can restart or restart the corresponding component service in rolling mode to restore the service. If the default policy is deleted by mistake, you can manually delete the service and then restart the component service.

Step 4 Choose **Access Manager** > **Reports** to view all security access policies of each component.

If there are many system policies, filter and search for policies by the policy name, policy type, component, resource, policy label, security zone, user, or user group. Alternatively, click **Export** to export related policies.



NOTE

- Generally, only one policy can be configured for a fixed resource object. If multiple policies are configured for the same resource object, the policies cannot be saved.
- For details about the priorities of different policies, see **Condition Priorities of the Ranger Permission Policy**.

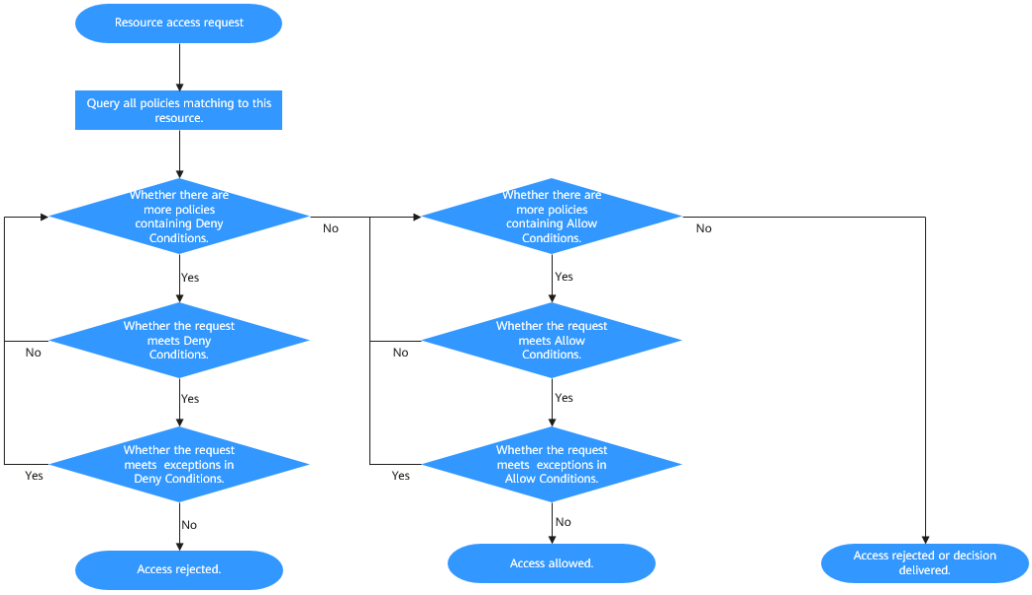
----End

Condition Priorities of the Ranger Permission Policy

When configuring a permission policy for a resource, you can configure Allow Conditions, Exclude from Allow Conditions, Deny Conditions, and Exclude from Deny Conditions for the resource, to meet unexpected requirements in different scenarios.

The priorities of different conditions are listed in descending order: Exclude from Deny Conditions > Deny Conditions > Exclude from Allow Conditions > Allow Conditions

The following figure shows the process of determining condition priorities. If the component resource request does not match the permission policy in Ranger, the system rejects the access by default. However, for HDFS and Yarn, the system delivers the decision to the access control layer of the component for determination.



For example, if you want to grant the read and write permissions of the **FileA** folder to the **groupA** user group, but the user in the group is not **UserA**, you can add an allowed condition and an exception condition.

12.21.4 Viewing Ranger Audit Information

The administrator can view audit logs of the Ranger running and the permission control after Ranger authentication is enabled on the Ranger web UI.

Viewing Ranger Audit Information

Step 1 Log in to the Ranger management page.

Step 2 Click **Audit** to view the audit information. For details about the content on each tab page, see **Table 12-322**. If there are a large number of items, click the search box and filter the items based on the keyword field.

Table 12-322 Audit information

Tab	Description
Access	Records audit information about users' access to component resources through Ranger authentication.
Admin	Records operation audit information on Ranger, such as the creation, update, and deletion of security access policies, component permission policies, and roles.
Login Sessions	Records session audit information for users who have logged in to Ranger.
Plugins	Records permission policy information of components in Ranger.
Plugin Status	Records audit information about permission policies of each component node.
User Sync	Records synchronized audit information of LDAP and Ranger users.

----End

12.21.5 Configuring a Security Zone

Security zone can be configured using Ranger. Administrators can divide resources of each component into multiple security zones where administrators set security policies for specified resources in the zones to facilitate management. Policies defined in a security zone apply only to resources in the zone. After service resources are allocated to the security zone, the access permission policies for the resources in the non-security zone do not take effect. The administrator of a security zone can set policies only in the security zone that the administrator belongs to.

Adding a Security Zone

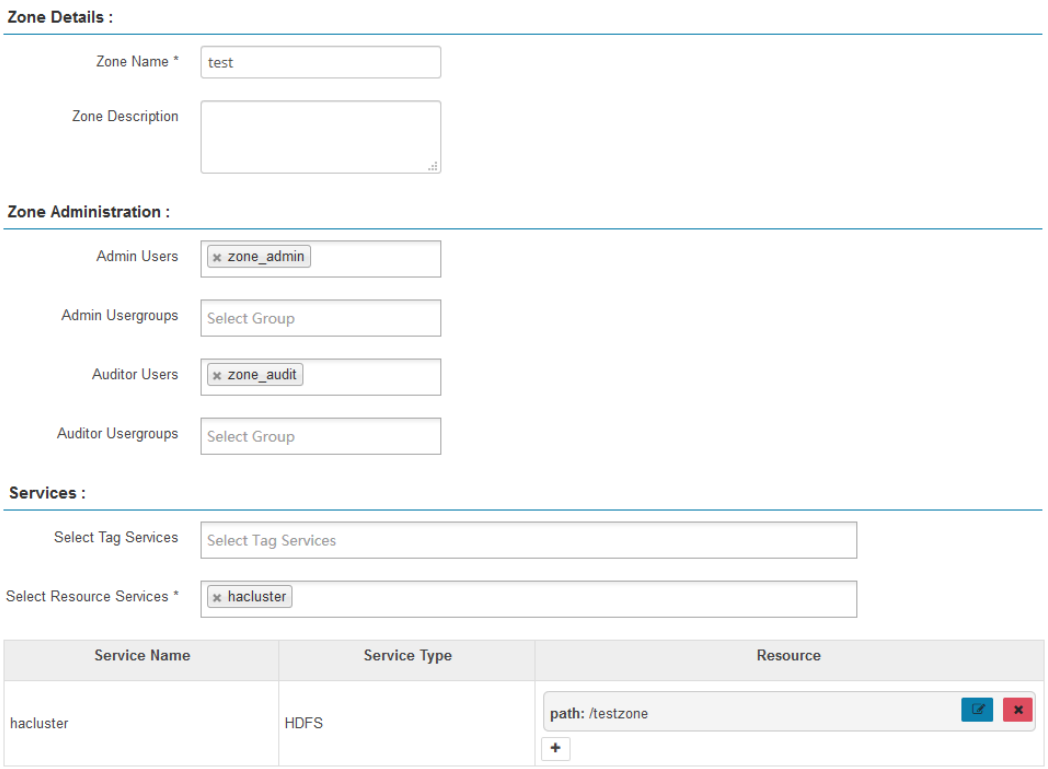
Step 1 Log in to the Ranger management page as the Ranger administrator.

Step 2 Click **Security Zone**. On the zone list page, click  to add a zone.

Table 12-323 Parameters for configuring a security zone

Parameter	Description	Example Value
Zone Name	Security zone	test
Zone Description	Description of the security zone	-
Admin Users/ Admin Usergroups	Management users and user groups in a security zone. You can add and modify permission policies for related resources in the security zone. At least one user or user group must be configured.	zone_admin
Auditor Users/ Auditor Usergroups	Audit users or user groups to be added. You can view the resource permission policies in the security zone. At least one user or user group must be configured.	zone_user
Select Tag Services	Tag information of a service	-
Select Resource Services	Services and resources in a security zone. After selecting a service, you need to add specific resource objects in the **Resource** column, such as the file directories of the HDFS server, Yarn queues, Hive databases and tables, and HBase tables and columns.	/testzone

For example, to create a security zone for the **/testzone** directory in HDFS, the configuration is as follows:



Step 3 Click **Save** and wait until the security zone is added successfully.

The Ranger administrator can view all security zones on the **Security Zone** page and click **Edit** to modify the attributes of a security zone. If resources do not need to be managed in a security zone, the Ranger administrator can click **Delete** to delete the security zone.

----End

Configuring Permission Policies in a Security Zone

Step 1 Log in to the Ranger management page as the administrator of a security zone.

Step 2 Select a security zone from the **Security Zone** drop-down list in the upper right corner of the Ranger home page to switch to the permission view of the security zone.

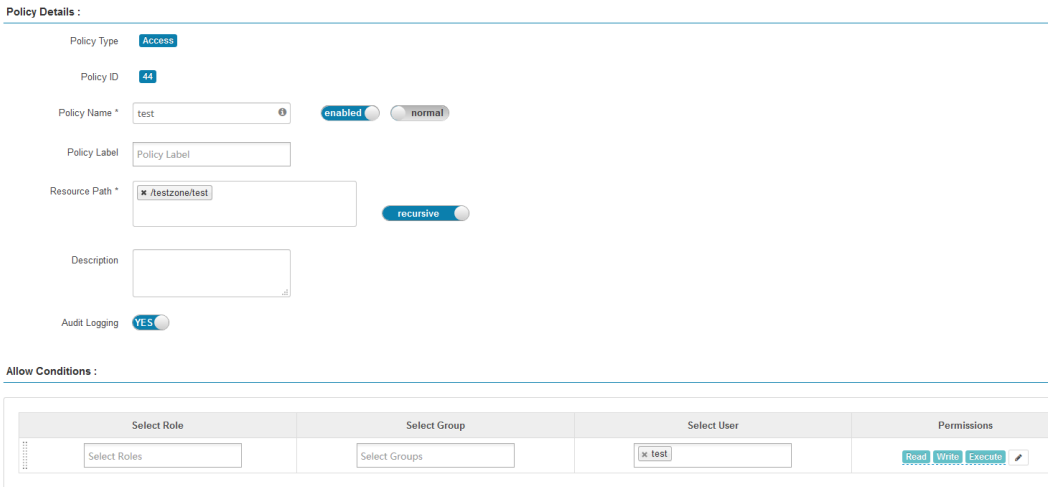
Step 3 Click the permission plug-in name of a component. The page for security access policy list of the component is displayed.

NOTE

In the policy list of each component, the default items generated by the system are automatically inherited to the security zone to ensure the permissions of some default users or user groups in the cluster.

Step 4 Click **Add New Policy** and configure resource access policies for related users or user groups based on the service scenario plan.

In this example, a policy that allows user test to access the **/testzone/test** directory is configured in the security zone.



The following access policies are examples for different components:

- **Adding a Ranger Access Permission Policy for HDFS**
- **Adding a Ranger Access Permission Policy for HBase**
- **Adding a Ranger Access Permission Policy for Hive**
- **Adding a Ranger Access Permission Policy for Yarn**
- **Adding a Ranger Access Permission Policy for Spark2x**
- **Adding a Ranger Access Permission Policy for Kafka**
- **Adding a Ranger Access Permission Policy for Storm**

After the policies are added, wait for about 30 seconds for them to take effect.

NOTE

- Policies defined in a security zone apply only to resources in the zone. After service resources are allocated to the security zone, the access permission policies for the resources in the non-security zone do not take effect.
- To configure access policies for resources outside the current security zone, click **Security Zone** in the upper right corner of the Ranger homepage to exit the current security zone.

----End

12.21.6 Changing the Ranger Data Source to LDAP for a Normal Cluster

By default, the Ranger data source of the security cluster can be accessed by FusionInsight Manager LDAP users. By default, the Ranger data source of a common cluster can be accessed by Unix users.

Prerequisites

- The cluster is in normal mode.
- The Ranger component has been installed.

Procedure

Step 1 Log in to the MRS console.

Step 2 Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to switch to the cluster details page.

Step 3 Click **Nodes** and select the node group whose **Node Type** is **Master**.

Step 4 Go to the ECS page of the active Master node and click **Remote Login**.

Step 5 Log in to a Master node as user **root**, go to the **/opt/Bigdata/components/FusionInsight_HD_8.1.0.1/Ranger** directory, and change the value of **ranger.usersync.sync.source** in the **configurations.xml** file to **ldap**.

```
ranger.usersync.sync.source
<value model="NoSec">ldap</value>
```

NOTE

Change the value of this parameter on all Master nodes.

Step 6 Run the following commands on the active Master node to restart the controller process:

su - omm

sh /opt/Bigdata/om-server_8.1.0.1/om/sbin/restart-controller.sh

NOTE

When the controller process is restarted, the MRS Manager page cannot be accessed for a short period of time, which is normal. After the controller process is restarted, you can access the MRS Manager page properly.

Step 7 Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > **Services** > **Ranger**. In the upper right corner of the **Dashboard** page, click **More** and choose **Synchronize Configuration**.

Step 8 On the Ranger instance page, select the **UserSync** instance and choose **More** > **Restart Instance**.

Step 9 On the **Dashboard** page of the Ranger service, click **RangerAdmin** and choose **Settings** > **Users/Groups/Roles** to check whether LDAP users exist.

----End

12.21.7 Viewing Ranger Permission Information

You can view Ranger permission settings, such as users, user groups, and roles.

Viewing Ranger Permission Information

Step 1 Log in to the Ranger management page as the administrator.

Step 2 Choose **Settings** > **Users**/**Groups**/**Roles** to view information about users, user groups, or roles in the system.

- Users: displays all user information synchronized from LDAP or OS to Ranger.
- Groups: displays information about all user groups and role information synchronized from LDAP or OS to Ranger.
- Roles: displays information about roles created in Ranger.

NOTE

- The users, roles, user groups created on FusionInsight Manager are automatically synchronized to Ranger periodically. The default period is 300,000 milliseconds (5 minutes). After roles and user groups in FusionInsight Manager are synchronized to Ranger, they become user groups. Only roles and user groups that are associated with users can be automatically synchronized to Ranger.
- The role created on the Ranger page is a set of users or user groups, which is used to flexibly set the permission access policies of components. The role is different from that on FusionInsight Manager.

----End

Adjusting Ranger User Types

Step 1 Log in to the Ranger management page.

To change the Ranger user type, you must log in as an **admin** user. For details about the user types, see **Ranger User Type**.

Step 2 Choose **Settings** > **Users**/**Groups**/**Roles**. In the list of users, click the name of the user whose type you want to change.

Step 3 Set **Select Role** to the type to be modified.

Step 4 Click **Save**.

----End

Creating a Ranger Role

The administrator can flexibly configure permission access policies for components based on users, user groups, or roles. User and user group information is automatically synchronized from LDAP, and roles can be manually added.

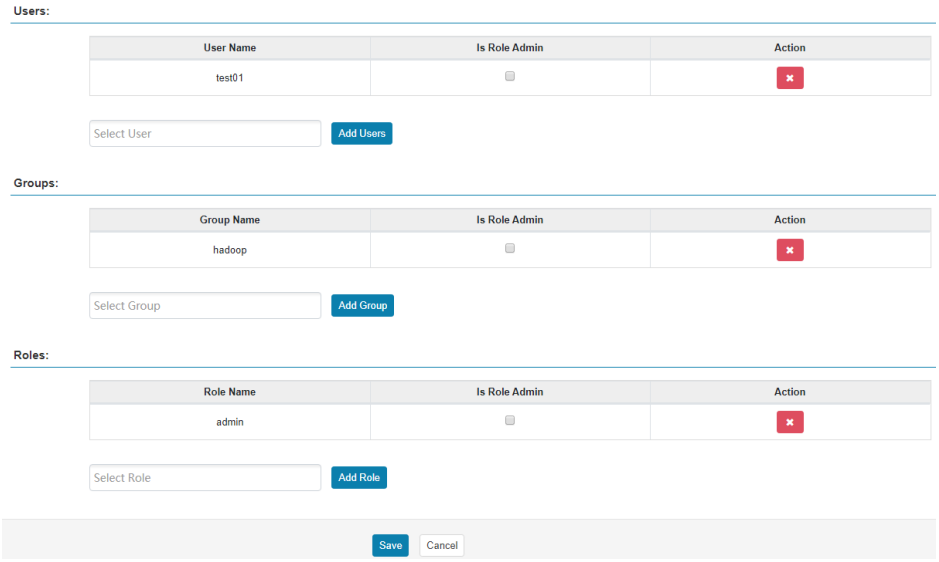
Step 1 Log in to the Ranger management page.

Step 2 Choose **Settings** > **Users**/**Groups**/**Roles** > **Roles** > **Add New Role**.

Step 3 Enter the role name and description as prompted.

Step 4 Add users, user groups, and sub-roles to the role.

- In the **Users** area, select a created user in the system and click **Add Users**.
- In the **Groups** area, select a created user group and click **Add Group**.
- In the **Roles** area, select a created role in the system and click **Add Role**.



Step 5 Click **Save**. The role is added successfully.

----End

12.21.8 Adding a Ranger Access Permission Policy for HDFS

Scenario

The administrator can use Ranger to configure the read, write, and execution permissions on HDFS directories or files for HDFS users.

Prerequisites

- The Ranger service has been installed and is running properly.
- You have created users, user groups, or roles for which you want to configure permissions.

Procedure

Step 1 Log in to the Ranger management page.

Step 2 On the homepage, click the component plug-in name in the **HDFS** area, for example, **hacluster**.

Step 3 Click **Add New Policy** to add an HDFS permission control policy.

Step 4 Configure the parameters listed in the table below based on the service demands.

Table 12-324 HDFS permission parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Resource Path	Resource path, which is the HDFS path folder or file to which the current policy applies. You can enter multiple values and use the wildcard (*), for example, **/test/***. To enable a subdirectory to inherit the permission of its upper-level directory, enable the recursion function. If recursion is enabled for the parent directory and a policy is configured for the subdirectory, the policy configured for the subdirectory is used. • **non-recursive**: recursion disabled • **recursive**: recursion enabled
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Permission and exception conditions allowed by a policy. The priority of an exception condition is higher than that of a normal condition. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add the corresponding permission. • **Read**: permission to read data • **Write**: permission to write data • **Execute**: execution permission • **Select/Deselect All**: Select or deselect all. If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users or user groups will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy. To add multiple permission control rules, click  . To delete a permission control rule, click  . Exclude from Allow Conditions: exception rules excluded from the allowed conditions
Deny All Other Accesses	Whether to reject all other access requests. • **True**: All other access requests are rejected. • **False**: **Deny Conditions** can be configured.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is the same as that of **Allow Conditions**. The priority of the rejection condition is higher than that of the allowed conditions configured in **Allow Conditions**. **Exclude from Deny Conditions**: exception rules excluded from the denied conditions

For example, to add the write permission for the **/user/test** directory of user **testuser**, the configuration is as follows:

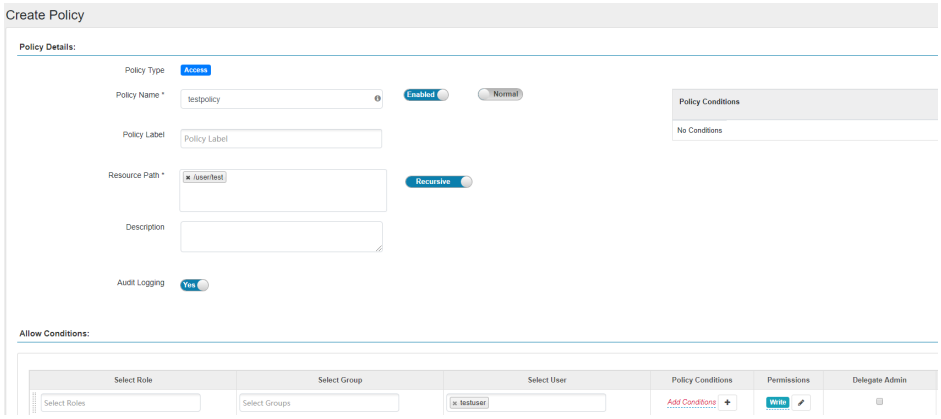


Table 12-325 Setting permissions

Task	Role Authorization
Setting the HDFS administrator permission	1. On the homepage, click the component plug-in name in the **HDFS** area, for example, **hacluster**. 2. Select the policy whose **Policy Name** is **all - path** and click  to edit the policy. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list.
Setting the permission for users to check and recover HDFS	1. Add a folder or a file path in **Resource Path**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Read** and **Execute**.
Setting the permission for users to read directories or files of other users	1. Add a folder or a file path in **Resource Path**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Read** and **Execute**.
Setting the permission for users to write data to files of other users	1. Add a folder or a file path in **Resource Path**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Write** and **Execute**.
Setting the permission for users to create or delete sub-files or sub-directories in the directory of other users	1. Add a folder or a file path in **Resource Path**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Write** and **Execute**.

Task	Role Authorization
Setting the permission for users to execute directories or files of other users	1. Add a folder or a file path in **Resource Path**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Execute**.
Setting the permission for allowing subdirectories to inherit all permissions of their parent directories	1. Add a folder or a file path in **Resource Path**. 2. Enable the recursion function. **Recursive** indicates that recursion is enabled.

Step 5 (Optional) Add the validity period of the policy. Click **Add Validity period** in the upper right corner of the page, set **Start Time** and **End Time**, and select **Time Zone**. Click **Save**. To add multiple policy validity periods, click  . To delete a policy validity period, click  .

Step 6 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

----End

12.21.9 Adding a Ranger Access Permission Policy for HBase

Scenario

Administrators can use Ranger to configure permissions on HBase tables, column families, and columns for HBase users.

Prerequisites

- The Ranger service has been installed and is running properly.
- You have created users, user groups, or roles for which you want to configure permissions.

Procedure

Step 1 Log in to the Ranger management page.

Step 2 On the home page, click the component plug-in name in the **HBASE** area, for example, **HBase**.

Step 3 Click **Add New Policy** to add an HBase permission control policy.

Step 4 Configure the parameters listed in the table below based on the service demands.

Table 12-326 HBase permission parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
HBase Table	Name of a table to which the policy applies. The value can contain wildcard (*). For example, **table1:*** indicates all tables in **table1**. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object. **NOTE** The value of **hbase.rpc.protection** of the HBase service plug-in on Ranger must be the same as that of **hbase.rpc.protection** on the HBase server. For details, see **When an HBase Policy Is Added or Modified on Ranger, Wildcard Characters Cannot Be Used to Search for Existing HBase Tables**.
HBase Column-family	Name of the column families to which the policy applies. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
HBase Column	Name of the column to which the policy applies. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Policy allowed condition. You can configure permissions and exceptions allowed by the policy. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add the corresponding permission. • **Read**: permission to read data • **Write**: permission to write data • **Create**: permission to create data • **Admin**: permission to manage data • **Select/Deselect All**: Select or deselect all. If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users or user groups will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy. To add multiple permission control rules, click  . To delete a permission control rule, click  . Exclude from Allow Conditions: policy exception conditions
Deny All Other Accesses	Whether to reject all other access requests. • **True**: All other access requests are rejected. • **False**: **Deny Conditions** can be configured.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is similar to that of **Allow Conditions**. The priority of **Deny Conditions** is higher than that of allowed conditions configured in **Allow Conditions**. **Exclude from Deny Conditions**: exception rules excluded from the denied conditions

Table 12-327 Setting permissions

Task	Role Authorization
Setting the HBase administrator permission	1. On the home page, click the component plug-in name in the **HBase** area, for example, **HBase**. 2. Select the policy whose **Policy Name** is **all - table, column-family, column** and click  to edit the policy. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list.
Setting the permission for users to create tables	1. In **HBase Table**, specify a table name. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Create**. 4. This user hase the following permissions: create table drop table truncate table alter table enable table flush table flush region compact disable enable desc
Setting the permission for users to write data to tables	1. In **HBase Table**, specify a table name. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Write**. 4. The user has the **put**, **delete**, **append**, **incr** and **bulkload** operation permissions.
Setting the permission for users to read data from tables	1. In **HBase Table**, specify a table name. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Read**. 4. This user hase the **get** and **scan** permissions.

Task	Role Authorization
Setting the permission for users to manage namespaces or tables	1. In **HBase Table**, specify a table name. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Admin**. 4. The user has the **rsgroup**, **peer**, **assign** and **balance** operation permissions.
Setting the permission for reading data from or writing data to columns	1. In **HBase Table**, specify a table name. 2. In **HBase Column-family**, specify the column family name. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Read** and **Write**.

NOTE

If a user performs the **desc** operation in **hbase shell**, the user must be granted the read permission on the **hbase:qouta** table.

Step 5 (Optional) Add the validity period of the policy. Click **Add Validity period** in the upper right corner of the page, set **Start Time** and **End Time**, and select **Time Zone**. Click **Save**. To add multiple policy validity periods, click  . To delete a policy validity period, click  .

Step 6 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

----End

12.21.10 Adding a Ranger Access Permission Policy for Hive

Scenario

The administrator can use Ranger to set permissions for Hive users. The default administrator account of Hive is **hive** and the initial password is **Hive@123**.

Prerequisites

- The Ranger service has been installed and is running properly.
- You have created users, user groups, or roles for which you want to configure permissions.

- The users must be added to the **hive** group.

Procedure

Step 1 Log in to the Ranger management page.

Step 2 On the home page, click the component plug-in name in the **HADOOP SQL** area, for example, **Hive**.

Step 3 On the **Access** tab page, click **Add New Policy** to add a Hive permission control policy.

Step 4 Configure the parameters listed in the table below based on the service demands.

Table 12-328 Hive permission parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**, **192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
database	Name of the Hive database to which the policy applies. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
table	Name of the Hive table to which the policy applies. To add a UDF-based policy, switch to UDF and enter the UDF name. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
Hive Column	Name of the column to which the policy applies. The value * indicates all columns. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Policy allowed condition. You can configure permissions and exceptions allowed by the policy. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add the corresponding permission. ● select: permission to query data ● update: permission to update data ● Create: permission to create data ● Drop: permission to drop data ● Alter: permission to alter data ● Index: permission to index data ● All: all permissions ● Read: permission to read data ● Write: permission to write data ● Temporary UDF Admin: temporary UDF management permission ● Select/Deselect All: Select or deselect all. To add multiple permission control rules, click . If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is similar to that of **Allow Conditions**.

Table 12-329 Setting permissions

Task	Role Authorization
role admin operation	1. On the home page, click **Settings** and choose **Roles**. 2. Click the role with **Role Name** set to **admin**. In the **Users** area, click **Select User** and select a username. 3. Click **Add Users**, select **Is Role Admin** in the row where the username is located, and click **Save**. **NOTE** Only user **rangeradmin** has the permission to access the **Settings** option on the Ranger page. After being bound to the Hive administrator role, perform the following operations during each maintenance operation: 1. Log in to the node where the Hive client is installed as the client installation user. 2. Run the following command to configure environment variables: For example, if the Hive client installation directory is **/opt/hiveclient**, run **source /opt/hiveclient/bigdata_env**. 3. Run the following command to authenticate the user: **kinit** *Hive service user* 4. Run the following command to log in to the client tool: **beeline** 5. Run the following command to update the administrator permissions: **set role admin;**
Creating a database table	1. Enter the policy name in **Policy Name**. 2. Enter or select the corresponding database on the right side of **database** and enter or select ***** on the right side of **column**. (To create a table, enter or select the corresponding table on the right side of **table**.) 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Create**.
Deleting a table	1. Enter the policy name in **Policy Name**. 2. Enter or select the corresponding database on the right side of **database** and enter and select ***** on the right side of **column**. (To delete a table, enter or select the corresponding table on the right side of **table**.) 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Drop**.

Task	Role Authorization
Query operation (**select**, **desc**, and **show**)	1. Enter the policy name in **Policy Name**. 2. Enter or select the corresponding database on the right side of **database** and enter or select ***** (***** indicates all columns) on the right side of **column**. (To create a table, enter or select the corresponding table on the right side of **table**.) 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **select**.
Alter operation	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right side of **database** and enter or select ***** on the right side of **column**. (For tables, enter or select the corresponding table on the right side of **table**.) 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Alter**.
LOAD operation	1. Enter the policy name in **Policy Name**. 2. On the right side of **database**, enter or select the corresponding database. On the right side of **table**, enter or select the corresponding table. On the right side of **column**, enter a column and select *****. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **update**.
INSERT and **DELETE** operations	1. Enter the policy name in **Policy Name**. 2. On the right side of **database**, enter or select the corresponding database. On the right side of **table**, enter or select the corresponding table. On the right side of **column**, enter a column and select *****. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **update**. 5. Configure the **submit** permission on the Yarn task queue. For details about how to configure the permission, see **Adding a Ranger Access Permission Policy for Yarn**.

Task	Role Authorization
GRANT/REVOKE operation	1. Enter the policy name in **Policy Name**. 2. On the right side of **database**, enter or select the corresponding database. On the right side of **table**, enter or select the corresponding table. On the right side of **column**, enter a column and select *****. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Select **Delegate Admin**.
ADD JAR operation	1. Enter the policy name in **Policy Name**. 2. Click **database**, and select **global** from the drop-down list. On the right of **global**, enter related information or select *****. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Temporary UDF Admin**.
UDF operation	1. Enter the policy name in **Policy Name**. 2. Enter or select the corresponding database on the right of **database**, and enter the corresponding udf function name on the right of **udf**. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select required permissions for the user (**udf** supports the **Create**, **select**, and **Drop** permissions).
VIEW operation	1. Enter the policy name in **Policy Name**. 2. On the right side of **database**, enter or select the corresponding database. On the right side of **table**, enter or select the corresponding table to be viewed. On the right side of **column**, enter a column and select *****. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select permissions for the user as required.
dfs command operation	The **dfs** operation can be performed only after you have run the **set role admin** command.
Operations on other user database tables	1. Perform the preceding operations to add the corresponding permissions. 2. Grant the read, write, and execution permissions on the HDFS paths of other user database tables to the user. For details, see **Adding a Ranger Access Permission Policy for HDFS**.

NOTE

- If you have specified an HDFS path when running commands, you need to be granted with the read, write, and execution permissions on the HDFS paths. For details, see **Adding a Ranger Access Permission Policy for HDFS**. You do not need to configure the Ranger policy of HDFS. You can use the Hive permission plug-in to add permissions to the role and assign the role to the corresponding user. If the HDFS Ranger policy can match the file or directory permission of the Hive database table, the HDFS Ranger policy is preferentially used.
- The URL policy in the Ranger policy is involved in the scenario where the Hive table is stored on OBS. Set the URL to the complete path of the object on OBS. The Read and Write permissions are used together with the URL. URL policies are not involved in other scenarios.
- The global policy in the Ranger policy is used only with the **Temporary UDF Admin** permission to control the upload of UDF packages.
- The **hiveservice** policy in the Ranger policy is used only with the **Service Admin** permission to control the permission to run the **kill query <queryId>** command to end the task that is being executed.
- The **lock**, **index**, **refresh**, and **replAdmin** permissions are not supported.
- Run the **show grant** command to view the table permission. The **grantor** column of the table **owner** is displayed as user **hive**. If the Ranger page is used or the **grant** command is used to grant permissions in the background, the **grantor** column is displayed as the corresponding user. To view the result of using the Hive permission plug-in, set **hive-ext.ranger.previous.privileges.enable** to **true** and run the **show grant** command.

Step 5 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

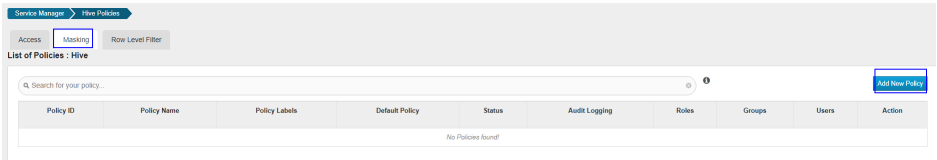
----End

Hive Data Masking

Ranger supports data masking for Hive data. It can process the returned result of the **select** operation you performed to mask sensitive information.

Step 1 Log in to the Ranger web UI. Click **Hive** in the **HADOOP SQL** area on the homepage.

Step 2 On the **Masking** tab page, click **Add New Policy** to add a Hive permission control policy.



Step 3 Configure the parameters listed in the table below based on the service demands.

Table 12-330 Hive data masking parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**, **192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Hive Database	Name of the Hive database to which the current policy applies.
Hive Table	Name of the Hive table to which the current policy applies.
Hive Column	Column name.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Mask Conditions	In the **Select Role**, **Select Group**, and **Select User** columns, select the object to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, then click **Add Permissions**, and select **select**. Click **Select Masking Option** and select a data masking policy. • Redact: Use **x** to mask all letters and **n** to mask all digits. • Partial mask: show last 4: Only the last four characters are displayed, and the rest characters are displayed using **x**. • Partial mask: show first 4: Only the first four characters are displayed, and the rest characters are displayed using **x**. • Hash: Replace the original value with the hash value. The Hive built-in function **mask_hash** is used. This is valid only for fields of the string, character, and varchar types. NULL is returned for fields of other types. • Nullify: Replace the original value with the NULL value. • Unmasked (retain original value): Keep the original value. • Date: show only year: Only the year part of the date string is displayed, and the default month and date start from January and Monday (**01/01**). • Custom: You customize policies using any valid return data type which is the same as the data type in the masked column. To add a multi-column masking policy, click  .

Step 4 Click **Add** to view the basic information about the policy in the policy list.

Step 5 After you perform the **select** operation on a table configured with a data masking policy on the Hive client, the system processes and displays the data.

NOTE

To process data, you must have the permission to submit tasks to the Yarn queue.

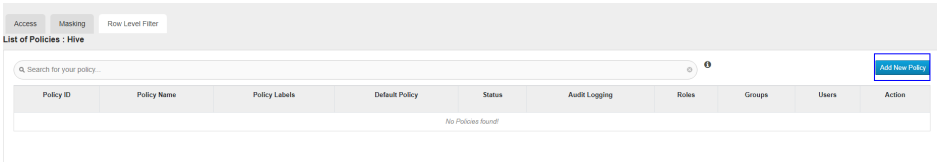
----End

Hive Row-Level Data Filtering

Ranger allows you to filter data at the row level when you perform the **select** operation on Hive data tables.

Step 1 Log in to the Ranger web UI. Click **Hive** in the **HADOOP SQL** area on the homepage.

Step 2 On the **Row Level Filter** tab page, click **Add New Policy** to add a row data filtering policy.



Step 3 Configure the parameters listed in the table below based on the service demands.

Table 12-331 Parameters for filtering Hive row data

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**, **192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Hive Database	Name of the Hive database to which the current policy applies.
Hive Table	Name of the Hive table to which the current policy applies.
Description	Policy description.
Audit Logging	Whether to audit the policy.
Row Filter Conditions	In the **Select Role**, **Select Group**, and **Select User** columns, select the object to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, then click **Add Permissions**, and select **Select**. Click **Row Level Filter** and enter data filtering rules. For example, if you want to filter the data in the **zhangsan** row in the **name** column of **table A**, the filtering rule is **name <>'zhangsan'**. For more information, see the official Ranger document. To add more rules, click  .

Step 4 Click **Add** to view the basic information about the policy in the policy list.

Step 5 After you perform the **select** operation on a table configured with a data masking policy on the Hive client, the system processes and displays the data.

NOTE

To process data, you must have the permission to submit tasks to the Yarn queue.

----End

12.21.11 Adding a Ranger Access Permission Policy for Yarn

Scenario

The administrator can use Ranger to configure Yarn administrator permissions for Yarn users, allowing them to manage Yarn queue resources.

Prerequisites

- The Ranger service has been installed and is running properly.
- You have created users, user groups, or roles for which you want to configure permissions.

Procedure

Step 1 Log in to the Ranger management page.

Step 2 On the home page, click the component plug-in name in the **YARN** area, for example, **Yarn**.

Step 3 Click **Add New Policy** to add a Yarn permission control policy.

Step 4 Configure the parameters listed in the table below based on the service demands.

Table 12-332 Yarn permission parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Queue	Queue name. The wildcard (*) is supported. To enable a sub-queue to inherit the permission of its upper-level queue, enable the recursion function. • **Non-recursive**: recursion disabled • **Recursive**: recursion enabled
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Policy allowed condition. You can configure permissions and exceptions allowed by the policy. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add the corresponding permission. • submit-app: permission to submit queue tasks • admin-queue: permission to manage queue tasks • Select/Deselect All: Select or deselect all. If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy. To add multiple permission control rules, click  . To delete a permission control rule, click  . Exclude from Allow Conditions: policy exception conditions
Deny All Other Accesses	Whether to reject all other access requests. • True: All other access requests are rejected. • **False**: **Deny Conditions** can be configured.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is similar to that of **Allow Conditions**. The priority of **Deny Conditions** is higher than that of allowed conditions configured in **Allow Conditions**. Exclude from Deny Conditions: exception rules excluded from the denied conditions

Table 12-333 Setting permissions

Task	Role Authorization
Setting the Yarn administrator permission	1. On the home page, click the component plug-in name in the **YARN** area, for example, **Yarn**. 2. Select the policy whose **Policy Name** is **all - queue** and click  to edit the policy. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list.

Task	Role Authorization
Setting the permission for a user to submit tasks in a specified Yarn queue	1. In **Queue**, specify a queue name. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **submit-app**.
Setting the permission for a user to manage tasks in a specified Yarn queue	1. In **Queue**, specify a queue name. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **admin-queue**.

Step 5 (Optional) Add the validity period of the policy. Click **Add Validity period** in the upper right corner of the page, set **Start Time** and **End Time**, and select **Time Zone**. Click **Save**. To add multiple policy validity periods, click  . To delete a policy validity period, click  .

Step 6 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

----End

NOTE

The permissions on Ranger Yarn are independent of each other. There is inclusion relationship among the permissions. Currently, the following permissions are supported:

- **submit-app**: permission to submit queue tasks
- **admin-queue**: permission to manage queue tasks

Although the **admin-queue** has the permission to submit tasks, it does not have the inclusion relationship with the **submit-app** permission.

12.21.12 Adding a Ranger Access Permission Policy for Spark2x

Scenario

The administrator can use Ranger to set permissions for Spark2x users.

NOTE

1. After Ranger authentication is enabled or disabled on Spark2x, you need to restart Spark2x.
2. Download the client again or manually update the client configuration file *Client installation directory*/**Spark2x/spark/conf/spark-defaults.conf**.

   Enable Ranger: **spark.ranger.plugin.authorization.enable=true**

   Disable Ranger: **spark.ranger.plugin.authorization.enable=false**
3. In Spark2x, spark-beeline (applications connected to JDBCServer) supports the Ranger IP address filtering policy (**Policy Conditions** in the Ranger permission policy), while spark-submit and spark-sql do not.

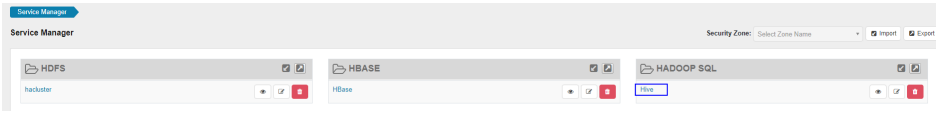
Prerequisites

- The Ranger service has been installed and is running properly.
- The Ranger authentication function of the Hive service has been enabled. After the Hive service is restarted, the Spark2x service is restarted.
- You have created users, user groups, or roles for which you want to configure permissions.
- The created user has been added to the **hive** user group.

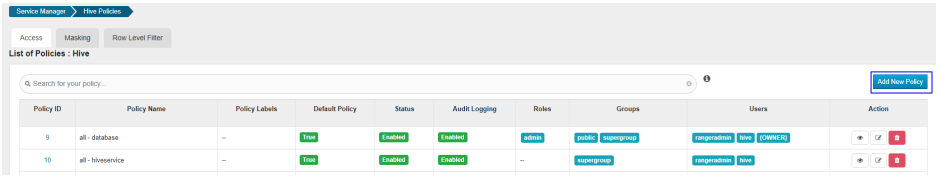
Procedure

Step 1 Log in to the Ranger management page.

Step 2 On the home page, click the component plug-in name in the **HADOOP SQL** area, for example, **Hive**.



Step 3 On the **Access** tab page, click **Add New Policy** to add a Spark2x permission control policy.



Step 4 Configure the parameters listed in the table below based on the service demands.

Table 12-334 Spark2x permission parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.

Parameter	Description
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
database	Name of the Spark2x database to which the policy applies. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
table	Name of the Spark2x table to which the policy applies. To add a UDF-based policy, switch to UDF and enter the UDF name. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
column	Name of the column to which the policy applies. The value ***** indicates all columns. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Policy allowed condition. You can configure permissions and exceptions allowed by the policy. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add the corresponding permission. ● select: permission to query data ● update: permission to update data ● Create: permission to create data ● Drop: permission to drop data ● Alter: permission to alter data ● Index: permission to index data ● All: all permissions ● Read: permission to read data ● Write: permission to write data ● Temporary UDF Admin: temporary UDF management permission ● Select/Deselect All: Select or deselect all. To add multiple permission control rules, click  . If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is similar to that of **Allow Conditions**.

Table 12-335 Setting permissions

Task	Operation
role admin operation	1. On the home page, click **Settings** and choose **Roles** > **Add New Role**. 2. Set **Role Name** to **admin**. In the **Users** area, click **Select User** and select a username. 3. Click **Add Users**, select **Is Role Admin** in the row where the username is located, and click **Save**. **NOTE** After being bound to the Hive administrator role, perform the following operations during each maintenance operation: 1. Log in to the node where the Hive client is installed as the client installation user. 2. Run the following command to configure environment variables: For example, if the Spark2x client installation directory is **/opt/client**, run **source /opt/client/bigdata_env**. 3. Run the following command to perform user authentication: **kinit Spark2x***Service user* 4. Run the following command to log in to the client tool: **spark-beeline** 5. Run the following command to update the administrator permissions: **set role admin;**
Creating a database table	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right of **database**. (If you want to create a database, enter the name of the database to be created or enter ***** to indicate a database with any name, and then select the name.) Enter and select the corresponding table name on the right of **table** and **column**. Wildcard characters (*) are supported. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Create**.

Task	Operation
Deleting a table	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right of **database**. (If you want to delete a database, enter the name of the database to be created or enter ***** to indicate a database with any name, and then select the name.) Enter and select the corresponding table name on the right of **table** and **column**. Wildcard characters (*) are supported. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Drop**. **NOTE** For CarbonData tables, only the owner of the corresponding database or table can perform the **drop** operation.
ALTER operation	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right of **database**, enter and select the corresponding table on the right of **table**, and enter and select the corresponding column name on the right of **column**. Wildcard characters (*) are supported. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Alter**.
LOAD operation	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right of **database**, enter and select the corresponding table on the right of **table**, and enter and select the corresponding column name on the right of **column**. Wildcard characters (*) are supported. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **update**.

Task	Operation
INSERT operation	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right of **database**, enter and select the corresponding table on the right of **table**, and enter and select the corresponding column name on the right of **column**. Wildcard characters (*) are supported. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **update**. 5. The user also needs to have the **submit-app** permission of the Yarn task queue. By default, the Hadoop user group has the **submit-app** permission of all Yarn task queues. For details about how to load a network instance to a cloud connection, see **Adding a Ranger Access Permission Policy for Yarn**.
GRANT operation	1. Enter the policy name in **Policy Name**. 2. Enter and select the corresponding database on the right of **database**, enter and select the corresponding table on the right of **table**, and enter and select the corresponding column name on the right of **column**. Wildcard characters (*) are supported. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Select **Delegate Admin**.
ADD JAR operation	1. Enter the policy name in **Policy Name**. 2. Click **database**, and select **global** from the drop-down list. On the right of **global**, enter related information and select ***. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Temporary UDF Admin**.

Task	Operation
VIEW and **INDEX** permissions	1. Enter the policy name in **Policy Name**. 2. On the right side of **database**, enter the database name and select the corresponding database. (If you want to delete a database, enter the database name and select *.) On the right side of **table**, enter a table name and select the view and index names. On the right side of **column**, enter a Hive column name, and select *. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select permissions for the user as required.
Operations on other user database tables	1. Perform the preceding operations to add the corresponding permissions. 2. Grant the read, write, and execution permissions on the HDFS paths of other user database tables to the current user. For details, see **Adding a Ranger Access Permission Policy for HDFS**.

NOTE

After Spark SQL access policy is added on Ranger, you need to add the corresponding path access policies in the HDFS access policy. Otherwise, data files cannot be accessed. For details, see **Adding a Ranger Access Permission Policy for HDFS**.

- The global policy in the Ranger policy is only used to associate with the **Temporary UDF Admin** permission to control the upload of UDF packages.
- When Ranger is used to control Spark SQL permissions, the **empower** syntax is not supported.

Step 5 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

----End

Data Masking of the Spark2x Table

Ranger supports data masking for Spark2x data. It can process the returned result of the **select** operation you performed to mask sensitive information.

Step 1 Log in to the Ranger WebUI and click the component plug-in name, for example, **Hive**, in the **HADOOP SQL** area on the home page.

Step 2 On the **Masking** tab page, click **Add New Policy** to add a Spark2x permission control policy.

Step 3 Configure the parameters listed in the table below based on the service demands.

Table 12-336 Spark2x data masking parameters

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Hive Database	Name of the Spark2x database to which the current policy applies.
Hive Table	Name of the Spark2x table to which the current policy applies.
Hive Column	Name of the Spark2x column to which the current policy applies.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Mask Conditions	In the **Select Group** and **Select User** columns, select the user group or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, then click **Add Permissions**, and select **select**. Click **Select Masking Option** and select a data masking policy. ● Redact: Use **x** to mask all letters and **n** to mask all digits. ● Partial mask: show last 4: Only the last four characters are displayed. ● Partial mask: show first 4: Only the first four characters are displayed. ● Hash: Perform hash calculation for data. ● Nullify: Replace the original value with the NULL value. ● Unmasked(retain original value): The original data is displayed. ● Date: show only year: Only the year information is displayed. ● Custom: You can use any valid Hive UDF (returns the same data type as the data type in the masked column) to customize the policy. To add a multi-column masking policy, click  .
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is similar to that of **Allow Conditions**.

----End

Spark2x Row-Level Data Filtering

Ranger allows you to filter data at the row level when you perform the **select** operation on Spark2x data tables.

Step 1 Log in to the Ranger WebUI and click the component plug-in name, for example, **Hive**, in the **HADOOP SQL** area on the home page.

Step 2 On the **Row Level Filter** tab page, click **Add New Policy** to add a row data filtering policy.

Step 3 Configure the parameters listed in the table below based on the service demands.

Table 12-337 Parameters for filtering Spark2x row data

Parameter	Description
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Hive Database	Name of the Spark2x database to which the current policy applies.
Hive Table	Name of the Spark2x table to which the current policy applies.
Description	Policy description.
Audit Logging	Whether to audit the policy.
Row Filter Conditions	In the **Select Role**, **Select Group**, and **Select User** columns, select the object to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, then click **Add Permissions**, and select **select**. Click **Row Level Filter** and enter data filtering rules. For example, if you want to filter the data in the **zhangsan** row in the **name** column of **table A**, the filtering rule is **name <>'zhangsan'**. For more information, see the official Ranger document. To add more rules, click  .

Step 4 Click **Add** to view the basic information about the policy in the policy list.

Step 5 After you perform the **select** operation on a table configured with a data masking policy on the Spark2x client, the system processes and displays the data.

----End

12.21.13 Adding a Ranger Access Permission Policy for Kafka

Scenario

The administrator can use Ranger to configure the read, write, and management permissions of the Kafka topic and the management permission of the cluster for the Kafka user. This section describes how to add the production permission of the **test** topic for the **test** user.

Prerequisites

- The Ranger service has been installed and is running properly.
- You have created users, user groups, or roles for which you want to configure permissions.

Procedure

Step 1 Log in to the Ranger management page.

Step 2 On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**.

Step 3 Click **Add New Policy** to add a Kafka permission control policy.

Step 4 Configure the following parameters based on the service demands.

Table 12-338 Kafka permission parameters

Parameter	Description
Policy Type	Access type.
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Name	Policy name, which can be customized and must be unique in the service.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
topic	Name of the topic applicable to the current policy. You can enter multiple values. The value can contain wildcards, such as **test**, **test***, and *. The **Include** policy applies to the current input object, and the **Exclude** policy applies to objects other than the current input object.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Permission and exception conditions allowed by a policy. The priority of an exception condition is higher than that of a normal condition. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which you want to assign permissions. Click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add corresponding permissions. ● Publish: production permission ● Consume: consumption permission ● Describe: query permission ● Create: topic creation permission ● Delete: topic deletion permission ● Describe Configs: configuration query permission ● Alter: permission to change the number of partitions of a topic. ● Alter Configs: configuration modification permission ● Select/Deselect All: Select or deselect all. To add multiple permission control rules, click  . If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is the same as that of **Allow Conditions**. The priority of the rejection condition is higher than that of the allowed conditions configured in **Allow Conditions**.

For example, to add the production permission for the **test** topic of user **testuser**, configure the following information:

Figure 12-43 Kafka permission parameters

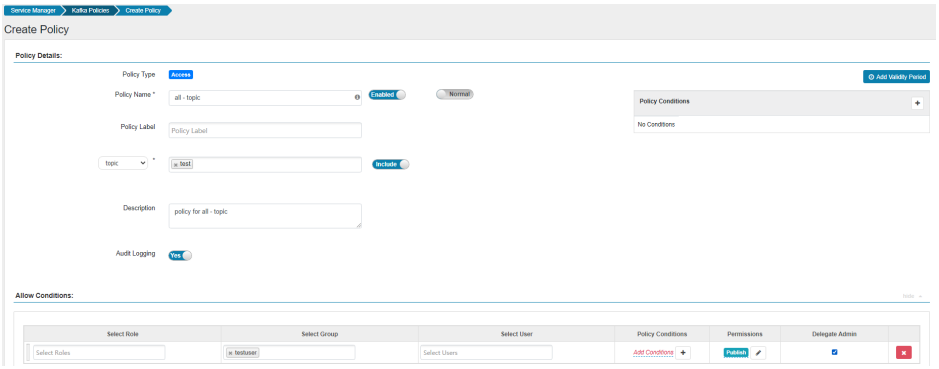


Table 12-339 Setting permissions

Scenario	Role Authorization
Setting the Kafka administrator permissions	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - topic** and click  to edit the policy. 3. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 4. Click **Add Permissions** and select **Select/ Deselect All**.
Setting the permission for a user to create a topic	1. Specify a topic name in **topic**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Create**. **NOTE** Currently, the Kafka kernel supports the **--zookeeper** and **--bootstrap-server** methods to create topics. The **--zookeeper** method will be deleted from the community in later versions. Therefore, you are advised to use the **--bootstrap-server** method to create topics. Note: Currently, Kafka supports only the authentication of topic creation in **--bootstrap-server** mode and does not support that in **--zookeeper** mode.

Scenario	Role Authorization
Setting the permission for a user to delete a topic	1. Specify a topic name in **topic**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Delete**. **NOTE** Currently, the Kafka kernel supports the **--zookeeper** and **--bootstrap-server** methods to delete topics. The **--zookeeper** method will be deleted from the community in later versions. Therefore, you are advised to use the **--bootstrap-server** method to delete topics. Note: Currently, Kafka supports only the authentication of topic deletion in **--bootstrap-server** mode and does not support that in **--zookeeper** mode.
Setting the permission for a user to query a topic	1. Specify a topic name in **topic**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Describe** and **Describe Configs**. **NOTE** Currently, the Kafka kernel supports the **--zookeeper** and **--bootstrap-server** methods to query topics. The **--zookeeper** method will be deleted from the community in later versions. Therefore, you are advised to use the **--bootstrap-server** method to query topics. Note: Currently, Kafka supports only the authentication of topic query in **--bootstrap-server** mode and does not support that in **--zookeeper** mode.
Setting the production permission of a user on a topic	1. Specify a topic name in **topic**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Publish**.
Setting the consumption permission of a user on a topic	1. Specify a topic name in **topic**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Consume**. **NOTE** During topic consumption, offset management is involved. Therefore, the **Consume** permission of **ConsumerGroup** must be enabled at the same time. For details, see **Setting a User's Permission to Submit ConsumerGroup Offsets**.
Setting the permission for a user to expand a topic (by adding partitions)	1. Specify a topic name in **topic**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Alter**.

Scenario	Role Authorization
Setting the permission for a user to modify the topic configuration	Currently, the Kafka kernel does not support to modify topic parameters based on **--bootstrap-server**. Therefore, Ranger does not support authentication for this behavior.
Setting all the management permissions of a user on a cluster	1. Enter a cluster name and select the cluster on the right side of **cluster**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Kafka Admin**.
Setting the permission for a user to create a cluster	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - cluster** and click  to edit the policy. 3. Enter a cluster name and select the cluster on the right side of **cluster**. 4. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 5. Click **Add Permissions** and select **Create**. **NOTE** The authentication of the **Create** operation of a cluster involves the following two scenarios: 1. After the **auto.create.topics.enable** parameter is enabled in the cluster, the client sends data to a topic that has not been created in the service. In this case, the system checks whether the user has the **Create** permission of the cluster. 2. If a user creates a large number of topics and is granted the **Cluster Create** permission, the user can create any topic in the cluster.
Setting the permission for a user to modify the cluster configuration	1. Enter a cluster name and select the cluster on the right side of **cluster**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Alter Configs**. **NOTE** The configuration modification permission allows you to modify the Broker and Broker Logger configurations. After the configuration modification permission is granted to a user, the user can query configuration details even if the user does not have the query permission. (The configuration modification permission includes the configuration query permission.)

Scenario	Role Authorization
Setting the permission for a user to query the cluster configuration	1. Enter a cluster name and select the cluster on the right side of **cluster**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Describe** and **Describe Configs**. **NOTE** You can only query Broker and Broker Logger information in the cluster, excluding topics.
Setting the Idempotent Write permission in a cluster for a user	1. Enter a cluster name and select the cluster on the right side of **cluster**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Idempotent Write**. **NOTE** This permission authenticates the **Idempotent Produce** behavior of the user's client.
Setting the permission to migrate partitions in a cluster for a user	1. Enter a cluster name and select the cluster on the right side of **cluster**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Alter**. **NOTE** The **Alter** permission of a cluster can be used to control permissions in the following scenarios: 1. In the **Partition Reassign** scenario, migrate the storage directory of replicas. 2. Elect a leader replica in each partition of the cluster. 3. Add or delete ACLs. Operations in scenarios **Step 4.1** and **Step 4.2** are between a controller and broker and between brokers in the cluster. When a cluster is created, this permission is granted to the built-in Kafka user by default. It is meaningless for a common user to be granted with this permission. Scenario **Step 4.3** involves the ACL management. ACLs are designed for authentication. Currently, Kafka authentication is hosted to Ranger. Therefore, this scenario is not involved (the configuration does not take effect).

Scenario	Role Authorization
Setting the Cluster Action permission in a cluster for a user	1. Enter a cluster name and select the cluster on the right side of **cluster**. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Cluster Action**. **NOTE** This permission controls the synchronization between the leader and follower replicas in the cluster and the communication between nodes. It has been granted to the built-in Kakfa user during cluster creation. It is meaningless for a common user to grant this permission.
Setting the TransactionalId permission for a user	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - transactionalid** and click  to edit the policy. 1. Set **transactionalid** to a transaction ID. 2. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 3. Click **Add Permissions** and select **Publish** and **Describe**. **NOTE** The **Publish** permission is used to authenticate client requests for which the transaction feature is enabled, for example, starting and ending a transaction, submitting an offset, and generating transactional data. The **Describe** permission is used to authenticate the requests from the client and coordinator that have enabled the transaction feature. If the transaction feature is enabled, you are advised to grant both the **Publish** and **Describe** permissions to users.

Scenario	Role Authorization
Setting the DelegationToken permission for a user	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - delegationtoken** and click  to edit the policy. 3. Set **delegationtoken** to a delegation token. 4. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 5. Click **Add Permissions** and select **Describe**. **NOTE** Currently, Ranger only controls the query permission of DelegationToken, but does not control its **create**, **renew**, and **expire** permissions.
Setting the permission for a user to query ConsumerGroup Offsets	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - consumergroup** and click  to edit the policy. 3. In **consumergroup**, configure the consumer group to be managed. 4. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 5. Click **Add Permissions** and select **Describe**.
Set the user's submission permission on **ConsumerGroup Offsets**.	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - consumergroup** and click  to edit the policy. 3. In **consumergroup**, configure the consumer group to be managed. 4. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 5. Click **Add Permissions** and select **Consume**. **NOTE** After a user is granted with the **Consume** permission of **ConsumerGroup**, the user is also granted with the **Describe** permission.

Scenario	Role Authorization
Setting the permission for a user to delete ConsumerGroup Offsets	1. On the home page, click the component plug-in name in the **KAFKA** area, for example, **Kafka**. 2. Select the policy whose **Policy Name** is **all - consumergroup** and click  to edit the policy. 3. In **consumergroup**, configure the consumer group to be managed. 4. In the **Allow Conditions** area, select a user from the **Select User** drop-down list. 5. Click **Add Permissions** and select **Delete**. **NOTE** When a user is granted with the **Delete** permission of **ConsumerGroup**, the user is also granted with the **Describe** permission.

Step 5 (Optional) Add the validity period of the policy. Click **Add Validity period** in the upper right corner of the page, set **Start Time** and **End Time**, and select **Time Zone**. Click **Save**. To add multiple policy validity periods, click  . To delete a policy validity period, click  .

Step 6 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

----End

12.21.14 Adding a Ranger Access Permission Policy for Storm

Scenario

The administrator can use Ranger to set permissions for Storm users.

Prerequisites

- The Ranger service has been installed and is running properly.
- You have created users, user groups, or roles for which you want to configure permissions.
- The Ranger authentication function has been enabled on the page. The option in the following figure controls whether to enable the Ranger plug-in for permission control. If the function is enabled, the Ranger authentication is used. Otherwise, the authentication mechanism of the component is used.

Procedure

Step 1 Log in to the Ranger web UI. Click **Storm** in the **STORM** area on the homepage.

Step 2 Click **Add New Policy** to add a Storm permission control policy.

Step 3 Configure the parameters listed in the table below based on the service demands.

Table 12-340 Storm permission parameters

Parameter	Description
Policy Conditions	IP address filtering policy, which can be customized. You can enter one or more IP addresses or IP address segments. The IP address can contain the wildcard character (*), for example, **192.168.1.10**,**192.168.1.20**, or **192.168.1.***.
Policy Name	Policy name, which can be customized and must be unique in the service. The **include** policy applies to the current input object, and the **exclude** policy applies to objects other than the current input object.
Policy Label	A label specified for the current policy. You can search for reports and filter policies based on labels.
Storm Topology	Name of the topology to which the current policy applies. One or more values can be entered.
Description	Policy description.
Audit Logging	Whether to audit the policy.

Parameter	Description
Allow Conditions	Policy allowed condition. You can configure permissions and exceptions allowed by the policy. In the **Select Role**, **Select Group**, and **Select User** columns, select the role, user group, or user to which the permission is to be granted, click **Add Conditions**, add the IP address range to which the policy applies, and click **Add Permissions** to add the corresponding permissions. • **Submit Topology**: Submit a topology. **NOTE** The Submit Topology permission takes effect only when **Storm Topology** is set to *****. • **File Upload**: Upload a file. • **File Download**: Download a file. • **Kill Topology**: Delete a topology. • **Rebalance**: Perform the rebalance operation. • **Activate**: Activate the topology permission. • **Deactivate**: Deactivate the topology permission. • **Get Topology Conf**: Obtain topology configurations. • **Get Topology**: Obtain a topology. • **Get User Topology**: Obtain user's topology. • **Get Topology Info**: Obtain topology information. • **Upload New Credential**: Upload a new credential. • **Select/Deselect All**: Select or deselect all. To add multiple permission control rules, click  . If users or user groups in the current condition need to manage this policy, select **Delegate Admin**. These users will become the agent administrators. The agent administrators can update and delete this policy and create sub-policies based on the original policy.
Deny Conditions	Policy rejection condition, which is used to configure the permissions and exceptions to be denied in the policy. The configuration method is similar to that of **Allow Conditions**.

Step 4 (Optional) Add the validity period of the policy. Click **Add Validity period** in the upper right corner of the page, set **Start Time** and **End Time**, and select **Time Zone**. Click **Save**. To add multiple policy validity periods, click  . To delete a policy validity period, click  .

Step 5 Click **Add** to view the basic information about the policy in the policy list. After the policy takes effect, check whether the related permissions are normal.

To disable a policy, click  to edit the policy and set the policy to **Disabled**.

If a policy is no longer used, click  to delete it.

----End

12.21.15 Ranger Log Overview

Log Description

Log path: The default storage path of Ranger logs is **/var/log/Bigdata/ranger/***Role name*.

- RangerAdmin: /var/log/Bigdata/ranger/rangeradmin (run logs)
- TagSync: /var/log/Bigdata/ranger/tagsync (run logs)
- UserSync: /var/log/Bigdata/ranger/usersync (run logs)

Log archive rule: The automatic compression and archive function is enabled for Ranger logs. By default, when the size of a log file exceeds 20 MB, the log file is automatically compressed. The naming rule of the compressed log file is as follows: *<Original log file name>-<yyyy-mm-dd_hh-mm-ss>.[ID]***.log.zip**. A maximum of 20 compressed file are retained.

Table 12-341 HDFS log list

Type	Name	Description
RangerAdmin run log file	access_log.*<DATE>*.log	Tomcat access log
	catalina.out	Tomcat service run log
	gc-worker.log	RangerAdmin garbage collection (GC) log
	postinstallDetail.log	Work log generated after an instance is started before installation
	prestartDetail.log	Log that records preparations before instance startup
	ranger-admin-*<hostname>*.log	RangerAdmin run log

Type	Name	Description
	ranger_admin_sql-*<hostname>*.log	RangerAdmin log used to retrieve DBService
	startDetail.log	Instance startup log
TagSync run log	cleanupDetail.log	Instance clearing log
	gc-worker.log	GC log file of an instance
	postinstallDetail.log	Work log generated after an instance is started before installation
	prestartDetail.log	Log that records preparations before instance startup
	ranger-tagsync-*<hostname>*.log	TagSync run log
	startDetail.log	Instance startup log
	tagsync.out	TagSync run log
UserSync run log	auth.log	UnixAuth service run log
	cleanupDetail.log	Instance clearing log
	gc-worker.log	GC log file of an instance
	postinstallDetail.log	Work log generated after an instance is started before installation
	prestartDetail.log	Log that records preparations before instance startup
	ranger-usersync-*<hostname>*.log	UserSync run log
	startDetail.log	Instance startup log

Log Levels

Table 12-342 describes the log levels provided by HDFS. The priorities of log levels are FATAL, ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-342 Log levels

Level	Description
FATAL	Logs of this level record fatal error information about the current event processing that may result in a system crash.
ERROR	Logs of this level record error information about the current event processing, which indicates that system running is abnormal.
WARN	Logs of this level record abnormal information about the current event processing. These abnormalities will not result in system faults.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > **Services** > **Ranger** > **Configurations**.

Step 3 Select **All Configurations**.

Step 4 On the menu bar on the left, select the log menu of the target role.

Step 5 Select a desired log level.

Step 6 Click **Save**. In the displayed dialog box, click **OK** to make the configuration take effect.

NOTE

The configurations take effect immediately without the need to restart the service.

----End

Log Formats

The following table lists the Ranger log formats.

Table 12-343 Log formats

Type	Format	Example Value
Run log	*<yyyy-MM-dd HH:mm:ss,SSS>*|*<Log level>*|*<Name of the thread that generates the log>*|*<Message in the log>*|*<Location where the log event occurs>*	2020-04-29 20:09:28,543 | INFO | http-bio-21401-exec-56 | Request comes from API call, skip cas filter. | CasAuthenticationFilter-Wrapper.java:25

12.21.16 Common Issues About Ranger

12.21.16.1 Why Ranger Startup Fails During the Cluster Installation?

Problem

During cluster installation, Ranger fails to be started, and the error message "ERROR: cannot drop sequence X_POLICY_REF_ACCESS_TYPE_SEQ " is displayed in the task list of the Manager process. How do I resolve this problem and properly install Ranger?

Answer

This issue may occur when two RangerAmdin instances are installed. If the instance installation fails, manually restart one RangerAdmin instance and then restart the other instance.

12.21.16.2 How Do I Determine Whether the Ranger Authentication Is Used for a Service?

Question

How do I determine whether the Ranger authentication is enabled for a service that supports the authentication?

Answer

Log in to FusionInsight Manager and choose **Cluster** > **Services** > *Name of the desired service*. On the service details page, click **More** and check whether the **Enable Ranger** option is available.

- If yes, the Ranger authentication plug-in is not enabled for the service. You can click **Enable Ranger** to enable the function.
- If no, the Ranger authentication plug-in has been enabled for the service. You can configure the permission policy for accessing the service resources on the Ranger management page.

12.21.16.3 Why Cannot a New User Log In to Ranger After Changing the Password?

Question

When a new user logs in to Ranger, why is the 401 error reported after the password is changed?

Answer

The UserSync synchronizes user data at an interval of 5 minutes by default. Therefore, a new user created on Manager cannot log in to the Ranger before the user data is successfully synchronized because the Ranger database does not have the user information. The user can log in to the Ranger only after the specified interval ends.

In non-security mode, the Ranger does not synchronize user data from Manager. Therefore, only the **admin** user can log in to the Ranger page.

12.21.16.4 When an HBase Policy Is Added or Modified on Ranger, Wildcard Characters Cannot Be Used to Search for Existing HBase Tables

Question

When a Ranger access permission policy is added for HBase and wildcard characters are used to search for an existing HBase table in the policy, the table cannot be found. The following error is reported in **/var/log/Bigdata/ranger/rangeradmin/ranger-admin-*log**:

```
Caused by: javax.security.sasl.SaslException: No common protection layer between client and server
at com.sun.security.sasl.gsskerb.GssKrb5Client.doFinalHandshake(GssKrb5Client.java:253)
at com.sun.security.sasl.gsskerb.GssKrb5Client.evaluateChallenge(GssKrb5Client.java:186)
at
org.apache.hadoop.hbase.security.AbstractHBaseSaslRpcClient.evaluateChallenge(AbstractHBaseSaslRpcClien
t.java:142)
at org.apache.hadoop.hbase.security.NettyHBaseSaslRpcClientHandler
$2.run(NettyHBaseSaslRpcClientHandler.java:142)
at org.apache.hadoop.hbase.security.NettyHBaseSaslRpcClientHandler
$2.run(NettyHBaseSaslRpcClientHandler.java:138)
at java.security.AccessController.doPrivileged(Native Method)
at javax.security.auth.Subject.doAs(Subject.java:422)
at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1761)
at
org.apache.hadoop.hbase.security.NettyHBaseSaslRpcClientHandler.channelRead0(NettyHBaseSaslRpcClientH
andler.java:138)
at
org.apache.hadoop.hbase.security.NettyHBaseSaslRpcClientHandler.channelRead0(NettyHBaseSaslRpcClientH
andler.java:42)
at
org.apache.hbase.thirdparty.io.netty.channel.SimpleChannelInboundHandler.channelRead(SimpleChannelInbo
undHandler.java:105)
at
org.apache.hbase.thirdparty.io.netty.channel.AbstractChannelHandlerContext.invokeChannelRead(AbstractCh
annelHandlerContext.java:362)
```

Answer

The value of **hbase.rpc.protection** of the HBase service plug-in on Ranger must be the same as that of **hbase.rpc.protection** on the HBase server.

Step 1 Log in to the Ranger management page. For details, see **Logging In to the Ranger Web UI**.

Step 2 In the **HBASE** area on the home page, click the component plug-in name, for example, the  button of HBase.

Step 3 Search for the configuration item **hbase.rpc.protection** and change its value to the value of **hbase.rpc.protection** on the HBase server.

Step 4 Click **Save**.

----End

12.22 Using Spark

12.22.1 Precautions

This section applies to versions earlier than MRS 3.x.

12.22.2 Getting Started with Spark

This section describes how to use Spark to submit a SparkPi job. SparkPi, a typical Spark job, is used to calculate the value of Pi (π).

Procedure

Step 1 Prepare the SparkPi program.

Multiple open-source Spark sample programs are provided, including SparkPi. Click **https://archive.apache.org/dist/spark/spark-2.1.0/spark-2.1.0-bin-hadoop2.7.tgz** to download the software package.

Decompress the software package to obtain the **spark-examples_2.11-2.1.0.jar** file, the sample program package, in the **spark-2.1.0-bin-hadoop2.7/examples/jars** directory. The **spark-examples_2.11-2.1.0.jar** sample program package contains the SparkPi program.

Step 2 Upload data to OBS.

1. Log in to OBS Console.
2. Choose **Parallel File System** > **Create Parallel File System** to create a file system named **sparkpi**.

   sparkpi is only an example. The file system name must be globally unique. Otherwise, the parallel file system fails to be created. Use the default values for other parameters.
3. Click the file system name **sparkpi** and click **Files**.
4. Click **Create Folder** to create the **program** folder..
5. Go to the **program** folder, click **Upload Object**, select the program package downloaded in **Step 1** from the local PC, and set **Storage Class** to **Standard**.

Step 3 Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click a cluster name.

Step 4 Submit the SparkPi job.

On the MRS console, click the **Jobs** tab and click **Create**. The **Create Job** page is displayed.

- Set **Type** to **SparkSubmit**.
- Set **Name** to **sparkPi**.
- Set **Program Path** to the path where programs are stored on OBS, for example, **obs://sparkpi/program/spark-examples_2.11-2.1.0.jar**.
- In **Program Parameter**, select **--class** for **Parameter** and set **Value** to **org.apache.spark.examples.SparkPi**.
- Set **Parameters** to **10**.
- Leave **Service Parameter** blank.

A job can be submitted only when the cluster is in the **Running** state.

After a job is submitted successfully, it is in the **Accepted** state by default. You do not need to manually execute the job.

Step 5 View the job execution result.

1. Go to the **Jobs** tab page and view job execution status.

   The job execution takes a while. After the jobs are complete, refresh the job list.

   Once a job has succeeded or failed, you cannot execute it again. However, you can add or copy a job, and set job parameters to submit a job again.

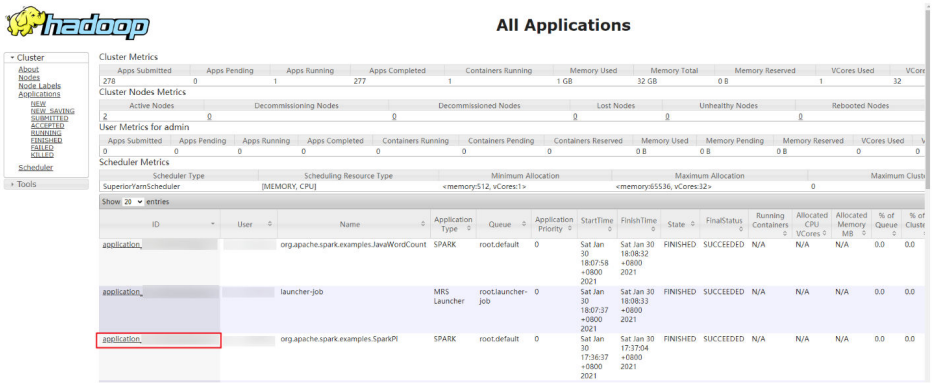
2. Go to the native Yarn page and view the job output information.

   a. On the **Jobs** tab page, locate the row that contains the target job and click **View Details** in the **Operation** column to obtain the actual job ID.

   b. Log in to Manager and choose **Services** > **Yarn** > **ResourceManager WebUI** > **ResourceManager (Active)**. The Yarn page is displayed.

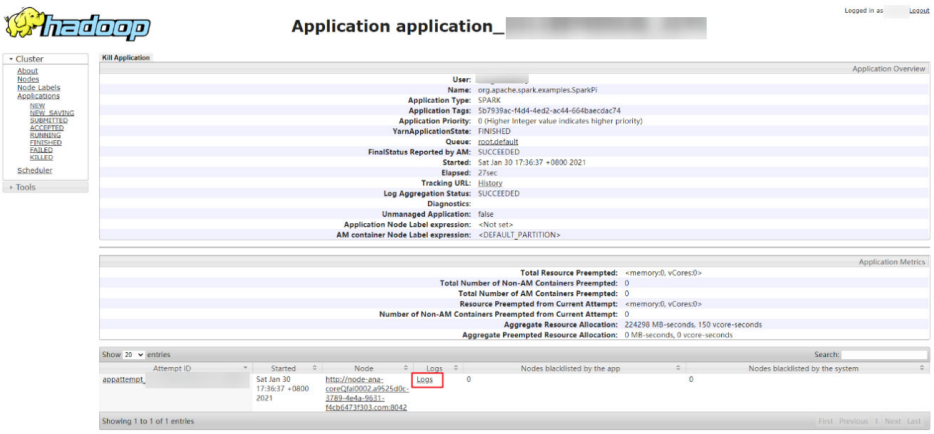
   c. Click the ID corresponding to the actual job ID.

   Figure 12-44 Yarn Web UI



   d. Click **Logs** in the job log area.

Figure 12-45 SparkPi job logs

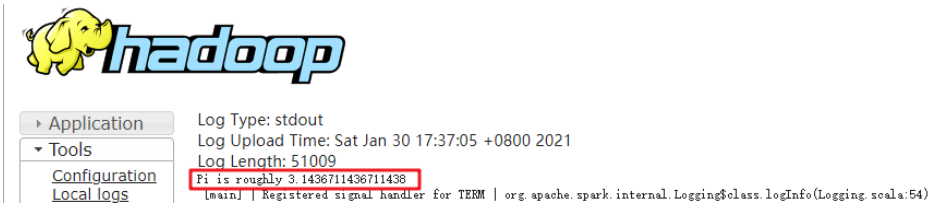


e. Click **here** to obtain more detailed logs.

Figure 12-46 More detailed logs of sparkPi jobs

f. Obtain the job execution result.

Figure 12-47 sparkPi job execution result



----End

12.22.3 Getting Started with Spark SQL

Spark provides the Spark SQL language that is similar to SQL to perform operations on structured data. This section describes how to use Spark SQL from scratch. Create a table named **src_data**, write a data record in each row of the table, and store the data in the **mrs_20160907** cluster. Then use SQL statements to query data in the table, and delete the table at last.

Prerequisites

You have obtained the AK/SK for writing data from an OBS data source to a Spark SQL table. To obtain it, perform as follows:

1. Log in to the management console.
2. Click the username and select **My Credentials** from the drop-down list.
3. On the displayed **My Credentials** page, click **Access Keys**.

4. Click **Create Access Key** to switch to the **Create Access Key** dialog box.
5. Enter the password and , and click **OK** to download the access key. Keep the access key secure.

Procedure

Step 1 Prepare data sources for Spark SQL analysis.

The sample text file is as follows:

```
abcd3ghji
efgh658ko
1234jjyu9
7h8kodfg1
kk99icxz3
```

Step 2 Upload data to OBS.

1. Log in to OBS Console.
2. Choose **Parallel File System** > **Create Parallel File System** to create a file system named **sparksql**.

   sparksql is only an example. The file system name must be globally unique. Otherwise, the parallel file system fails to be created.
3. Click the name of the **sparksql** file system and click **Files**.
4. Click **Create Folder** to create the **input** folder.
5. Go to the **input** folder, choose **Upload File** > **add file**, select the local TXT file, and click **Upload**.

Step 3 Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click a cluster name.

Step 4 Import the text file from OBS to HDFS.

1. Click the **Files** tab.
2. On the **HDFS File List** tab page, click **Create Folder**, and create a folder named **userinput**.
3. Go to the **userinput** folder, and click **Import Data**.
4. Select the OBS and HDFS paths and click **OK**.

   OBS Path: **obs://sparksql/input/sparksql-test.txt**

   HDFS Path: **/user/userinput**

Step 5 Submit the SQL statement.

1. On the MRS console, select **Job Management**.

   A job can be submitted only when the **mrs_20160907** cluster is in the **Running** state.
2. Enter the Spark SQL statement for table creation.

   When entering Spark SQL statements, ensure that the statement characters are not more than 10,000.

   Syntax:

   CREATE [EXTERNAL] **TABLE** [IF NOT EXISTS] *table_name* [(col_name data_type [COMMENT col_comment], ...)] [COMMENT table_comment] [PARTITIONED **BY** (col_name data_type [COMMENT col_comment], ...)]

[CLUSTERED **BY** (col_name, col_name, ...) [SORTED **BY** (col_name [ASC|DESC], ...)] INTO num_buckets BUCKETS] [ROW FORMAT row_format] [STORED **AS** file_format] [LOCATION hdfs_path];

You can use the following two methods to create a table example:

- Method 1: Create table **src_data** and write data in every row.
  - The data source is stored in the folder of HDFS: ***create external table*** *src_data* ***(line string) row format delimited fields terminated by '||n' stored as textfile location*** *'/user/userinput*';
  - The data source is stored in the **/sparksql/input** folder of OBS: **create external table** *src_data* **(line string) row format delimited fields terminated by '\\n' stored as textfile location** *'obs://AK:SK@sparksql/input*';

    For details about how to obtain the AK/SK, see **Prerequisites**.
- Method 2: Create table **src_data1** and load data to the table in batches.

  create table *src_data1* **(line string) row format delimited fields terminated by ','** ;

  load data inpath *'/user/userinput/sparksql-test.txt'* **into table** *src_data1*;

  NOTE

  When method 2 is used, the data from OBS cannot be loaded to the created tables directly.

3. Enter the Spark SQL statement for table query.

   Syntax:

   SELECT col_name **FROM** *table_name*;

   Example of querying all data in the **src_data** table:

   select * from src_data;

4. Enter the Spark SQL statement for table deletion.

   Syntax:

   DROP TABLE [IF EXISTS] *table_name*;

   Example of deleting the **src_data** table:

   drop table src_data;

5. Click **Check** to check the statement correctness.
6. Click **OK**.

   After the Spark SQL statements are submitted, the statement execution results are displayed in the result column.

Step 6 Delete the cluster.

----End

12.22.4 Using the Spark Client

After an MRS cluster is created, you can create and submit jobs on the client. The client can be installed on nodes inside or outside the cluster.

- Nodes inside the cluster: After an MRS cluster is created, the client has been installed on the master and core nodes in the cluster by default. For details, see . Then, log in to the node where the MRS client is installed..
- Nodes outside the cluster: You can install the client on nodes outside a cluster. For details about how to install a client, see , and log in to the node where the MRS client is installed..

Using the Spark Client

Step 1 Based on the client location, log in to the node where the client is installed. For details, see , or .

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the Spark shell command. The following provides an example:

spark-beeline

----End

12.22.5 Accessing the Spark Web UI

The Spark web UI is used to view the running status of Spark applications. Google Chrome is recommended for better user experience.

Spark has two web UIs.

- Spark UI: used to display the status of running applications.

  The UI includes the following parts: Jobs, Stages, Storage, Environment, Executors, SQL, and JDBC/ODBC Server. The Streaming application has the Streaming tab in addition to the preceding parts.
- History Server UI: used to display the status of Spark applications that are complete or incomplete.

  The UI includes the application ID, application name, start time, end time, execution time, and owner information.

Spark UI

Step 1 Access the component management page.

- For versions earlier than MRS 3.*x*, click the cluster name to go to the cluster details page and choose **Components**.

NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 2 Select **Yarn**. In the **Yarn Summary** area, click **ResourceManager** in **ResourceManager Web UI** to access the web UI.

Step 3 Locate the Spark application. Click **ApplicationMaster** in the last column of the application information. The Spark UI is displayed.

Figure 12-48 ApplicationMaster

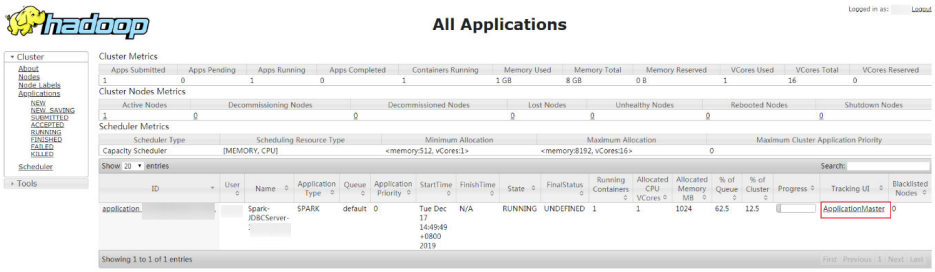


Figure 12-49 Spark UI

----End

History Server

Step 1 Access the component management page.

- For versions earlier than MRS 3.*x*, click the cluster name to go to the cluster details page and choose **Components**.

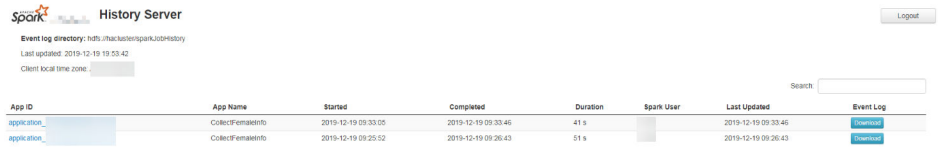
NOTE

If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 2 Select **Spark**. In the **Spark Summary** area, click **JobHistory** corresponding to **Spark Web UI** to access the web UI.

Figure 12-50 Spark History Server



----End

12.22.6 Interconnecting Spark with OpenTSDB

12.22.6.1 Creating a Table and Associating It with OpenTSDB

Function

MRS Spark can be used to access the data source of OpenTSDB, create and associate tables in the Spark, and query and insert the OpenTSDB data.

Use the **CREATE TABLE** command to create a table and associate it with an existing metric in OpenTSDB.

NOTE

If no metric exists in OpenTSDB, an error will be reported when the corresponding table is queried.

Syntax

```
CREATE TABLE [IF NOT EXISTS] OPENTSDB_TABLE_NAME   USING OPENTSDB OPTIONS (
'metric' = 'METRIC_NAME',
'tags' = 'TAG1,TAG2'
);
```

Keyword

Parameter	Description
metric	Indicates the name of the metric in OpenTSDB corresponding to the table to be created.
tags	Indicates the tags corresponding to the metric. The tags are used for classification, filtering, and quick retrieval. You can set 1 to 8 tags, which are separated by commas (,). The parameter value includes values of all tagKs in the corresponding metric.

Precautions

When creating a table, you do not need to specify the **timestamp** and **value** fields. The system automatically builds the following fields based on the specified tags. The fields **TAG1** and **TAG2** are specified by tags.

- TAG1 String

- TAG2 String
- timestamp Timestamp
- value double

Example

Create table **opentsdb_table** and associate it with metric **city.temp** of the OpenTSDB component.

```
CREATE table opentsdb_table using opentsdb OPTIONS ('metric'='city.temp',  'tags'='city,location');
```

12.22.6.2 Inserting Data to the OpenTSDB Table

Function

Run the **INSERT INTO** statement to insert the data in the table to the associated OpenTSDB metric.

Syntax

```
INSERT INTO TABLE_NAME SELECT * FROM SRC_TABLE;
INSERT INTO TABLE_NAME VALUES(XXX);
```

Keyword

Parameter	Description
TABLE_NAME	Indicates the name of the associated OpenTSDB table.
SRC_TABLE	Indicates the name of the table from which data is obtained. This parameter can be set to a name of a common table.

Precautions

- The inserted data cannot be **null**. If the inserted data is the same as the original data or only the **value** is different, the inserted data overwrites the original data.
- **INSERT OVERWRITE** is not supported.
- You are advised not to concurrently insert data into a table. If you concurrently insert data into a table, there is a possibility that conflicts occur, leading to data insertion failures.
- The **TIMESTAMP** format supports only yyyy-MM-dd hh:mm:ss.

Example

Insert data into table **opentsdb_table**.

```
insert into opentsdb_table values('city1','city2','2018-05-03 00:00:00',21);
```

12.22.6.3 Querying an OpenTSDB Table

This **SELECT** command is used to query data in an OpenTSDB table.

Syntax

```
SELECT * FROM table_name WHERE tagk=tagv LIMIT number;
```

Keyword

Parameter	Description
LIMIT	Used to limit the query results.
number	Only the INT type is supported.

Precautions

- The to-be-queried table must exist. Otherwise, an error is reported.
- The value of **tagv** must exist. Otherwise, an error occurs.

Example

Query data in the **opentsdb_table** table.

```
SELECT * FROM opentsdb_table LIMIT 100;
SELECT * FROM opentsdb_table WHERE city='city1';
```

12.22.6.4 Modifying the Default Configuration Data

By default, OpenTSDB connects to the local TSD process of the node where the Spark executor resides. In MRS, use the default configuration.

Table 12-344 OpenTSDB data source configuration

Parameter	Description	Example Value
spark.sql.datasource.opentsdb.host	Indicates the IP address of the connected TSD process.	Null (default value) **xx.xx.xx.xx** indicates the IP address. Separate multiple IP addresses with commas (,).
spark.sql.datasource.opentsdb.port	Indicates the port number of the TSD process.	4242 (default value)

spark.sql.datasource.opentsdb.randomSeed	Indicates whether to use the random seed when the **spark.sql.datasource.opentsdb.host** is set to multiple addresses. If this parameter is set to **false**, all executors on the same node are connected to the same host. In this way, **spark.blacklist.enabled=true** can be used to implement task fault tolerance.	false (default value)

Example

Run the **set** statement in **spark-sql** and **spark-beeline**, and then run other SQL statements.

```
set spark.sql.datasource.opentsdb.host = 192.168.2.143,192.168.2.158;
SELECT * FROM opentsdb_table ;
```

12.23 Using Spark2x

12.23.1 Precautions

This section applies to MRS 3.*x* or later.

12.23.2 Basic Operation

12.23.2.1 Getting Started

This section describes how to use Spark2x to submit Spark applications, including Spark Core and Spark SQL. Spark Core is the kernel module of Spark. It executes tasks and is used to compile Spark applications. Spark SQL is a module that executes SQL statements.

Scenario Description

Develop a Spark application to perform the following operations on logs about netizens' dwell time for online shopping on a weekend.

- Collect statistics on female netizens who dwell on online shopping for more than 2 hours on the weekend.
- The first column in the log file records names, the second column records genders, and the third column records the dwell durations in the unit of minute. Three columns are separated by comma (,).

log1.txt: logs collected on Saturday

```
LiuYang,female,20
YuanJing,male,10
GuoYijun,male,5
CaiXuyu,female,50
Liyuan,male,20
FangBo,female,50
LiuYang,female,20
YuanJing,male,10
GuoYijun,male,50
CaiXuyu,female,50
FangBo,female,60
```

log2.txt: logs collected on Sunday

```
LiuYang,female,20
YuanJing,male,10
CaiXuyu,female,50
FangBo,female,50
GuoYijun,male,5
CaiXuyu,female,50
Liyuan,male,20
CaiXuyu,female,50
FangBo,female,50
LiuYang,female,20
YuanJing,male,10
FangBo,female,50
GuoYijun,male,50
CaiXuyu,female,50
FangBo,female,60
```

Prerequisites

- On Manager, you have created a user and granted the HDFS, Yarn, Kafka, and Hive permissions to the user.
- You have installed and configured tools such as IntelliJ IDEA and JDK based on the development language.
- You have installed the Spark2x client and configured the client network connection.
- For Spark SQL programs, you have started Spark SQL or Beeline on the client to enter SQL statements.

Procedure

Step 1 Obtain the sample project and import it to IDEA. Import the JAR package on which the sample project depends. Use IDEA to configure and generate JAR packages.

Step 2 Prepare the data required by the sample project.

Save the original log files in the scenario description to the HDFS system.

1. Create two text files (**input_data1.txt** and **input_data2.txt**) on the local host and copy the content in the **log1.txt** and **log2.txt** files to the **input_data1.txt** and **input_data2.txt** files, respectively.
2. Create the **/tmp/input** directory in HDFS, and upload **input_data1.txt** and **input_data2.txt** to the **/tmp/input** directory:

Step 3 Upload the generated JAR package to the Spark2x running environment (Spark2x client), for example, **/opt/female**.

Step 4 Go the client directory, configure the environment variables, and log in to the system. When you use a client to connect to a specific instance in a scenario where multiple Spark2x instances are installed or Spark and Spark2x instances are installed, run the following commands to load the environment variables of the instance.

source bigdata_env

source Spark2x/component_env

kinit <*service user for authentication***>**

Step 5 Run the following script in the **bin** directory to submit the Spark application:

spark-submit --class *com.xxx.bigdata.spark.examples.FemaleInfoCollection* ***--master yarn-client*** */opt/female/FemaleInfoCollection.jar <inputPath>*

NOTE

- **FemaleInfoCollection.jar** is the JAR package generated in **Step 1**.
- **<inputPath>** is the directory created in **Step 2.2**.

Step 6 (Optional) After calling the **spark-sql** or **spark-beeline** script in the **bin** directory, directly enter SQL statements to perform operations such as query.

For example, create a table, insert a piece of data, and then query the table.

```
spark-sql> CREATE TABLE TEST(NAME STRING, AGE INT);
Time taken: 0.348 seconds
spark-sql>INSERT INTO TEST VALUES('Jack', 20);
Time taken: 1.13 seconds
spark-sql> SELECT * FROM TEST;
Jack	20
Time taken: 0.18 seconds, Fetched 1 row(s)
```

Step 7 View the running result of the Spark application.

- View the running result data in a specified file.

  The storage path and format of the result data are specified by the Spark application.

- Check the running status on the web page.

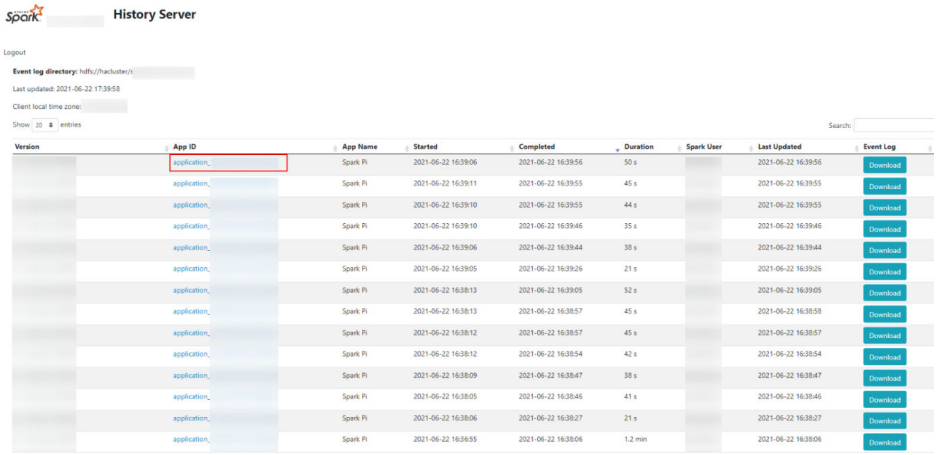
  a. Log in to Manager. Select **Spark2x** from the **Service** drop-down list.

  b. Go to the Spark2x overview page and click an instance in the Spark web UI, for example, **JobHistory2x(host2)**.

  c. The History Server UI is displayed.

  The History Server UI is used to display the status of Spark applications that are complete or incomplete.

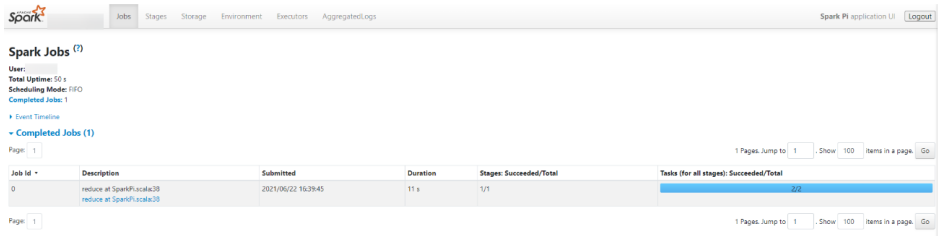
Figure 12-51 History Server UI



d. Select an application ID and click this page to go to the Spark UI of the application.

Spark UI: used to display the status of running applications.

Figure 12-52 Spark UI



- View Spark logs to learn application runtime conditions.

  View **Spark2x Logs** to learn application running status, and adjust applications based on log information.

----End

12.23.2.2 Configuring Parameters Rapidly

Overview

This section describes how to quickly configure common parameters and lists parameters that are not recommended to be modified when Spark2x is used.

Common parameters to be configured

Some parameters have been adapted during cluster installation. However, the following parameters need to be adjusted based on application scenarios. Unless otherwise specified, the following parameters are configured in the **spark-defaults.conf** file on the Spark2x client.

Table 12-345 Common parameters to be configured

Configuration Item	Description	Default Value
spark.sql.parquet.compression.codec	Used to set the compression format of a non-partitioned Parquet table. Set the queue in the **spark-defaults.conf** configuration file on the JDBCServer server.	snappy
spark.dynamicAllocation.enabled	Indicates whether to use dynamic resource scheduling, which is used to adjust the number of executors registered with the application according to scale. Currently, this parameter is valid only in Yarn mode. The default value for JDBCServer is **true**, and that for the client is **false**.	false
spark.executor.memory	Indicates the memory size used by each executor process. Its character sting is in the same format as the JVM memory (example: 512 MB or 2 GB).	4G
spark.sql.autoBroadcastJoinThreshold	Indicates the maximum value for the broadcast configuration when two tables are joined. ● When the size of a field in a table involved in an SQL statement is less than the value of this parameter, the system broadcasts the SQL statement. ● If the value is set to **-1**, broadcast is not performed.	10485760
spark.yarn.queue	Specifies the Yarn queue where JDBCServer resides. Set the queue in the **spark-defaults.conf** configuration file on the JDBCServer server.	default
spark.driver.memory	In a large cluster, you are advised to configure the memory used by the 32 GB to 64 GB driver process, that is, the SparkContext initialization process (for example, 512 MB and 2 GB).	4G

Configuration Item	Description	Default Value
spark.yarn.security.credentials.hbase.enabled	Indicates whether to enable the function of obtaining HBase tokens. If the Spark on HBase function is required and a security cluster is configured, set this parameter to **true**. Otherwise, set this parameter to **false**.	false
spark.serializer	Used to serialize the objects that are sent over the network or need to be cached. The default value of Java serialization applies to any Serializable Java object, but the running speed is slow. Therefore, you are advised to use **org.apache.spark.serializer.KryoSerializer** and configure Kryo serialization. It can be any subclass of **org.apache.spark.serializer.Serializer**.	org.apache.spark.serializer.JavaSerializer
spark.executor.cores	Indicates the number of kernels used by each executor. Set this parameter in standalone mode and Mesos coarse-grained mode. When there are sufficient kernels, the application is allowed to execute multiple executable programs on the same worker. Otherwise, each application can run only one executable program on each worker.	1
spark.shuffle.service.enabled	Indicates a long-term auxiliary service in NodeManager for improving shuffle computing performance.	false
spark.sql.adaptive.enabled	Indicates whether to enable the adaptive execution framework.	false
spark.executor.memoryOverhead	Indicates the heap memory to be allocated to each executor, in MB. This is the memory that occupies the overhead of the VM, similar to the internal string and other built-in overhead. The value increases with the executor size (usually 6% to 10%).	1 GB
spark.streaming.kafka.direct.lifo	Indicates whether to enable the LIFO function of Kafka.	false

Parameters Not Recommended to Be Modified

The following parameters have been adapted during cluster installation. You are not advised to modify them.

Table 12-346 Parameters not recommended to be modified

Configuration Item	Description	Default Value or Configuration Example
spark.password.factory	Selects the password parsing mode.	org.apache.spark.om.util.FIPasswordFactory
spark.ssl.ui.protocol	Sets the SSL protocol of the UI.	TLSv1.2
spark.yarn.archive	Archives Spark JAR files, which are distributed to Yarn cache. If this parameter is set, the value will replace **<code> spark.yarn.jars </code>** and be archived in the containers of all applications. The archive should contain the JAR files in its root directory. Archives can also be hosted on HDFS to speed up file distribution.	hdfs://hacluster/user/spark2x/jars/8.1.0.1/spark-archive-2x.zip **NOTE** The version 8.1.0.1 is used as an example. Replace it with the actual version number.
spark.yarn.am.extraJavaOptions	Indicates a string of extra JVM options to pass to the YARN ApplicationMaster in client mode. Use **spark.driver.extraJavaOptions** in cluster mode.	-Dlog4j.configuration=./__spark_conf__/__hadoop_conf__/log4j-executor.properties -Djava.security.auth.login.config=./__spark_conf__/__hadoop_conf__/jaas-zk.conf -Dzookeeper.server.principal=zookeeper/hadoop.<*system domain name*> -Djava.security.krb5.conf=./__spark_conf__/__hadoop_conf__/kdc.conf -Djdk.tls.ephemeralDHKeySize=2048
spark.shuffle.servicev2.port	Indicates the port for the shuffle service to monitor requests for obtaining data.	27338

Configuration Item	Description	Default Value or Configuration Example
spark.ssl.historyServer.enabled	Sets whether the history server uses SSL.	true
spark.files.overwrite	When the target file exists and its content does not match that of the source file, whether to overwrite the file added through **SparkContext.addFile()**.	false
spark.yarn.cluster.driver.extraClassPath	Indicates the extraClassPath of the driver in Yarn-cluster mode. Set the parameter to the path and parameters of the server.	${BIGDATA_HOME}/common/runtime/security
spark.driver.extraClassPath	Indicates the extra class path entries attached to the class path of the driver.	${BIGDATA_HOME}/common/runtime/security
spark.yarn.dist.innerfiles	Sets the files that need to be uploaded to HDFS from Spark in Yarn mode.	/Spark_path/spark/conf/s3p.file,/Spark_path/spark/conf/locals3.jceks *Spark_path* is the installation path of the Spark client.
spark.sql.bigdata.register.dialect	Registers the SQL parser.	org.apache.spark.sql.hbase.HBaseSQLParser
spark.shuffle.manager	Indicates the data processing mode. There are two implementation modes: sort and hash. The sort shuffle has a higher memory utilization. It is the default option in Spark 1.2 and later versions.	SORT

Configuration Item	Description	Default Value or Configuration Example
spark.deploy.zookeeper.url	Indicates the address of ZooKeeper. Multiple addresses are separated by commas (,).	For example: host1:2181,host2:2181,host3:2181
spark.broadcast.factory	Indicates the broadcast mode.	org.apache.spark.broadcast.TorrentBroadcastFactory
spark.sql.session.state.builder	Session state constructor.	org.apache.spark.sql.hive.FIHiveACLSessionStateBuilder
spark.executor.extraLibraryPath	Sets the special library path used when the executor JVM is started.	${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-Hadoop-3.1.1/hadoop/lib/native
spark.ui.customErrorPage	Indicates whether to display the custom error information page when an error occurs on the page.	true
spark.httpdProxy.enable	Indicates whether to use the httpd proxy.	true
spark.ssl.ui.enabledAlgorithms	Sets the SSL algorithm of UI.	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,TLS_DHE_DSS_WITH_AES_256_GCM_SHA384,TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,TLS_DHE_DSS_WITH_AES_128_GCM_SHA256
spark.ui.logout.enabled	Sets the logout button for the web UI of the Spark component.	true
spark.security.hideInfo.enabled	Indicates whether to hide sensitive information on the UI.	true

Configuration Item	Description	Default Value or Configuration Example
spark.yarn.cluster.driver.extraLibraryPath	Indicates the **extraLibraryPath** of the driver in Yarn-cluster mode. Set this parameter to the path and parameters of the server.	${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/FusionInsight-Hadoop-3.1.1/hadoop/lib/native
spark.driver.extraLibraryPath	Sets a special library path for starting the driver JVM.	${DATA_NODE_INSTALL_HOME}/hadoop/lib/native
spark.ui.killEnabled	Allows stages and jobs to be stopped on the web UI.	true
spark.yarn.access.hadoopFileSystems	Spark can access multiple NameService instances. If there are multiple NameService instances, set this parameter to all the NameService instances and separate them with commas (,).	hdfs://hacluster,hdfs://hacluster

Configuration Item	Description	Default Value or Configuration Example
spark.yarn.cluster.driver.extraJavaOptions	Indicates extra JVM option passed to the executor, for example, GC setting and logging. Do not set Spark attributes or heap size using this option. Instead, set Spark attributes using the SparkConf object or the **spark-defaults.conf** file specified when the spark-submit script is called. Set heap size using **spark.executor.memory**.	-Xloggc:<LOG_DIR>/gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=10M -Dlog4j.configuration=./__spark_conf__/__hadoop_conf__/log4j-executor.properties -Djava.security.auth.login.config=./__spark_conf__/__hadoop_conf__/jaas-zk.conf -Dzookeeper.server.principal=zookeeper/hadoop.*<system domain name>* -Djava.security.krb5.conf=./__spark_conf__/__hadoop_conf__/kdc.conf -Djetty.version=x.y.z -Dorg.xerial.snappy.tempdir=${BIGDATA_HOME}/tmp/spark2x_app -Dcarbon.properties.filepath=./__spark_conf__/__hadoop_conf__/carbon.properties -Djdk.tls.ephemeralDHKeySize=2048
spark.driver.extraJavaOptions	Indicates a series of extra JVM options passed to the driver,	-Xloggc:${SPARK_LOG_DIR}/indexserver-omm-%p-gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:MaxDirectMemorySize=512M -XX:MaxMetaspaceSize=512M -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=10M -XX:OnOutOfMemoryError='kill -9 %p' -Djetty.version=x.y.z -Dorg.xerial.snappy.tempdir=${BIGDATA_HOME}/tmp/spark2x/JDBCServer/snappy_tmp -Djava.io.tmpdir=${BIGDATA_HOME}/tmp/spark2x/JDBCServer/io_tmp -Dcarbon.properties.filepath=${SPARK_CONF_DIR}/carbon.properties -Djdk.tls.ephemeralDHKeySize=2048 -Dspark.ssl.keyStore=${SPARK_CONF_DIR}/child.keystore #{java_stack_prefer}
spark.eventLog.overwrite	Indicates whether to overwrite any existing file.	false

Configuration Item	Description	Default Value or Configuration Example
spark.eventLog.dir	Indicates the directory for logging Spark events if **spark.eventLog.enabled** is set to **true**. In this directory, Spark creates a subdirectory for each application and logs events of the application in the subdirectory. You can also set a unified address similar to the HDFS directory so that the History Server can read historical files.	hdfs://hacluster/spark2xJobHistory2x
spark.random.port.min	Sets the minimum random port.	22600
spark.authenticate	Indicates whether Spark authenticates its internal connections. If the application is not running on Yarn, see **spark.authenticate.secret**.	true
spark.random.port.max	Sets the maximum random port.	22899
spark.eventLog.enabled	Indicates whether to log Spark events, which are used to reconstruct the web UI after the application execution is complete.	true

Configuration Item	Description	Default Value or Configuration Example
spark.executor.extraJavaOptions	Indicates extra JVM option passed to the executor, for example, GC setting and logging. Do not set Spark attributes or heap size using this option.	-Xloggc:<LOG_DIR>/gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=10M -Dlog4j.configuration=./log4j-executor.properties -Djava.security.auth.login.config=./jaas-zk.conf -Dzookeeper.server.principal=zookeeper/hadoop.*<system domain name>* -Djava.security.krb5.conf=./kdc.conf -Dcarbon.properties.filepath=./carbon.properties -Xloggc:<LOG_DIR>/gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=10M -Dlog4j.configuration=./__spark_conf__/__hadoop_conf__/log4j-executor.properties -Djava.security.auth.login.config=./__spark_conf__/__hadoop_conf__/jaas-zk.conf -Dzookeeper.server.principal=zookeeper/hadoop.*<system domain name>* -Djava.security.krb5.conf=./__spark_conf__/__hadoop_conf__/kdc.conf -Dcarbon.properties.filepath=./__spark_conf__/__hadoop_conf__/carbon.properties -Djdk.tls.ephemeralDHKeySize=2048
spark.sql.authorization.enabled	Indicates whether to enable authentication for the Hive client.	true

12.23.2.3 Common Parameters

Overview

This section describes common configuration items used in Spark. Subsections are divided by feature so that you can quickly find required configuration items. If you

use MRS clusters, most parameters described in this section have been adapted and you do not need to configure them again. For details about the parameters that need to be configured based on the site requirements, see **Configuring Parameters Rapidly**.

Configuring the Number of Stage Retries

When FetchFailedException occurs in a Spark task, a stage retry is triggered. To prevent infinite stage retries, the number of stage retries is limited. The number of retry times can be adjusted based on the site requirements.

Configure the following parameters in the **spark-defaults.conf** file on the Spark client.

Table 12-347 Parameter description

Parameter	Description	Default Value
spark.stage.maxConsecutiveAttempts	Indicates the maximum number of stage retries.	4

Configuring Whether to Use Cartesian Product

To enable the Cartesian product function, configure the following parameter in the **spark-defaults.conf** configuration file of Spark.

Table 12-348 Cartesian product parameters

Parameter	Description	Default Value
spark.sql.crossJoin.enabled	Indicates whether to allow implicit Cartesian product execution. • **true**: Implicit Cartesian product execution is allowed. • **false**: Implicit Cartesian product execution is not allowed. In this case, only CROSS JOIN can be explicitly included in the query.	true

NOTE

- For JDBC applications, configure this parameter in the **spark-defaults.conf** configuration file of the server.
- For tasks submitted by the Spark client, configure this parameter in the **spark-defaults.conf** configuration file of the client.

Configuring Security Authentication for Long-Time Spark Tasks

In security mode, if the **kinit** command is used for security authentication when the Spark CLI (such as spark-shell, spark-sql, or spark-submit) is used, the task fails due to authentication expiration when the task is running for a long time.

Set the following parameters in the **spark-defaults.conf** configuration file on the client. After the configuration is complete, run the Spark CLI again.

NOTE

If this parameter is set to **true**, ensure that the values of **keytab** and **principal** in **spark-defaults.conf** and **hive-site.xml** are the same.

Table 12-349 Parameter description

Parameter	Description	Default Value
spark.kerberos.principal	Indicates the principal user who has the Spark operation permission. Contact the administrator to obtain the principal user.	-
spark.kerberos.keytab	Indicates the name and path of the keytab file used to configure Spark operation permissions. Contact the administrator to obtain the keytab file.	-
spark.security.bigdata.loginOnce	Indicates whether the principal user logs in to the system only once. **true**: single login; **false**: multiple logins. The difference between a single login and multiple logins is as follows: The Spark community uses the Kerberos user to log in to the system for multiple times. However, the TGT or token may expire, causing the application to fail to run for a long time. The Kerberos login mode of DataSight is modified to allow users to log in only once, which effectively resolves the expiration problem. The restrictions are as follows: The principal and keytab configuration items of Hive must be the same as those of Spark. **NOTE** If this parameter is set to **true**, ensure that the values of **keytab** and **principal** in **spark-defaults.conf** and **hive-site.xml** are the same.	true

Python Spark

Python Spark is the third programming language of Spark except Scala and Java. Different from Java and Scala that run on the JVM platform, Python Spark has its own Python process as well as the JVM process. The following configuration items

apply only to Python Spark scenarios. However, other configuration items can also take effect in Python Spark scenarios.

Table 12-350 Parameter description

Parameter	Description	Default Value
spark.python.profile	Indicates whether to enable profiling on the Python worker. Use **sc.show_profiles()** to display the analysis results or display the analysis results before the Driver exits. You can use **sc.dump_profiles(path)** to dump the results to a disk. If some analysis results have been manually displayed, they will not be automatically displayed before the driver exits. By default, **pyspark.profiler.BasicProfiler** is used. You can transfer the specified profiler during SparkContext initialization to overwrite the default profiler.	false
spark.python.worker.memory	Indicates the memory size that can be used by each Python worker process during aggregation. The value format is the same as that of the specified JVM memory, for example, 512 MB and 2 GB. If the memory used by a process during aggregation exceeds the value of this parameter, data will be written to disks.	512m
spark.python.worker.reuse	Indicates whether to reuse Python workers. If the reuse function is enabled, a fixed number of Python workers will be reused by the next batch of submitted tasks instead of forking a Python process for each task. This function is useful in large-scale broadcasting because the data does not need to be transferred from the JVM to the Python workers again for the next batch of submitted tasks.	true

Dynamic Allocation

Dynamic resource scheduling is a unique feature of the On Yarn mode. This function can be used only after Yarn External Shuffle is enabled. When Spark is used as a resident service, dynamic resource scheduling greatly improves resource utilization. For example, the JDBCServer process does not accept JDBC requests in most of the time. Therefore, releasing resources in this period greatly reduces the waste of cluster resources.

Table 12-351 Parameter description

Parameter	Description	Default Value
spark.dynamicAllocation.enabled	Indicates whether to use dynamic resource scheduling, which is used to adjust the number of executors registered with the application according to scale. Currently, this parameter is valid only in Yarn mode. To enable dynamic resource scheduling, set **spark.shuffle.service.enabled** to **true**. Related parameters are as follows: **spark.dynamicAllocation.minExecutors**, **spark.dynamicAllocation.maxExecutors**, and **spark.dynamicAllocation.initialExecutors**.	• JDBCServer2x: true • SparkResource2x: false
spark.dynamicAllocation.minExecutors	Indicates the minimum number of executors.	0
spark.dynamicAllocation.initialExecutors	Indicates the number of initial executors.	spark.dynamicAllocation.minExecutors
spark.dynamicAllocation.maxExecutors	Indicates the maximum number of executors.	2048
spark.dynamicAllocation.schedulerBacklogTimeout	Indicates the first timeout period for scheduling. The unit is second.	1s
spark.dynamicAllocation.sustainedSchedulerBacklogTimeout	Indicates the second and later timeout interval for scheduling.	1s
spark.dynamicAllocation.executorIdleTimeout	Indicates the idle timeout interval for common executors. The unit is second.	60

Parameter	Description	Default Value
spark.dynamicAllocation.cachedExecutorIdleTimeout	Indicates the idle timeout interval for executors with cached blocks.	• JDBCServer2x: 2147483647s • IndexServer2x: 2147483647s • SparkResource2x: 120

Spark Streaming

Spark Streaming is a streaming data processing function provided by the Spark batch processing platform. It processes data input from external systems in **mini-batch** mode.

Configure the following parameters in the **spark-defaults.conf** file on the Spark client.

Table 12-352 Parameter description

Parameter	Description	Default Value
spark.streaming.receiver.writeAheadLog.enable	Indicates whether to enable the write-ahead log (WAL) function. After this function is enabled, all input data received by the receiver is saved in the WAL. WAL ensures that data can be restored if the driver program becomes faulty.	false
spark.streaming.unpersist	Determines whether to automatically remove RDDs generated and saved by Spark Streaming from the Spark memory. If this function is enabled, original data received by Spark Streaming is also automatically cleared. If this function is disabled, original data and RDDs cannot be automatically cleared. External applications can access the data in Streaming. This, however, occupies more Spark memory resources.	true

Spark Streaming Kafka

The receiver is an important component of Spark Streaming. It receives external data, encapsulates the data into blocks, and provides the blocks for Streaming to

consume. The most common data source is Kafka. Spark Streaming integrates Kafka to ensure reliability and can directly use Kafka as the RDD input.

Table 12-353 Parameter description

Parameter	Description	Default Value
spark.streaming.kafka.maxRatePerPartition	Indicates the maximum rate (number of records per second) for reading data from each Kafka partition if the Kafka direct stream API is used.	-
spark.streaming.blockInterval	Indicates the interval (ms) for accumulating data received by a Spark Streaming receiver into a data block before the data is stored in Spark. A minimum value of 50 ms is recommended.	200ms
spark.streaming.receiver.maxRate	Indicates the maximum rate (number of records per second) for each receiver to receive data. The value **0** or a negative value indicates no limit to the rate.	-
spark.streaming.receiver.writeAheadLog.enable	Indicates whether to use ReliableKafkaReceiver. This receiver ensures the integrity of streaming data.	false

Netty/NIO and Hash/Sort Configuration

Shuffle is critical for big data processing, and the network is critical for the entire shuffle process. Currently, Spark supports two shuffle modes: hash and sort. There are two network modes: Netty and NIO.

Table 12-354 Parameter description

Parameter	Description	Default Value
spark.shuffle.manager	Indicates the data processing mode. There are two implementation modes: sort and hash. The sort shuffle has a higher memory utilization. It is the default option in Spark 1.2 and later versions.	SORT

Parameter	Description	Default Value
spark.shuffle.consolidateFiles	(Only in hash mode) To merge intermediate files created during shuffle, set this parameter to **true**. Decreasing the number of files to be created can improve the processing performance of the file system and reduce risks. If the **ext4** or **xfs** file system is used, you are advised to set this parameter to **true**. Due to file system restrictions, this setting on **ext3** may reduce the processing performance of a server with more than eight cores.	false
spark.shuffle.sort.bypassMergeThreshold	This parameter is valid only when **spark.shuffle.manager** is set to **sort**. When Map aggregation is not performed and the number of partitions for Reduce tasks is less than or equal to the value of this parameter, do not merge and sort data to prevent performance deterioration caused by unnecessary sorting.	200
spark.shuffle.io.maxRetries	(Only in Netty mode) If this parameter is set to a non-zero value, fetch failures caused by I/O-related exceptions will be automatically retried. This retry logic helps the large shuffle keep stable when long GC pauses or intermittent network disconnections occur.	12
spark.shuffle.io.numConnectionsPerPeer	(Only in Netty mode) Connections between hosts are reused to reduce the number of connections between large clusters. For a cluster with many disks but a few hosts, this function may make concurrent requests unable to occupy all disks. Therefore, you can increase the value of this parameter.	1
spark.shuffle.io.preferDirectBufs	(Only in Netty mode) The off-heap buffer is used to reduce GC during shuffle and cache block transfer. In an environment where off-heap memory is strictly limited, you can disable it to force all applications from Netty to use heap memory.	true
spark.shuffle.io.retryWait	(Only in Netty mode) Specifies the duration for waiting for fetch retry, in seconds. The maximum delay caused by retry is **maxRetries** x **retryWait**. The default value is 15 seconds.	5

Common Shuffle Configuration

Table 12-355 Parameter description

Parameter	Description	Default Value
spark.shuffle.spill	If this parameter is set to **true**, data is overflowed to the disk to limit the memory usage during a Reduce task.	true
spark.shuffle.spill.compress	Indicates whether to compress the data overflowed during shuffle. The algorithm specified by **spark.io.compression.codec** is used for data compression.	true
spark.shuffle.file.buffer	Specifies the size of the memory buffer for storing output streams of each shuffle file, in KB. These buffers can reduce the number of disk seek and system calls during the creation of intermediate shuffle file streams. You can also set this parameter by setting **spark.shuffle.file.buffer.kb**.	32KB
spark.shuffle.compress	Indicates whether to compress the output files of a Map task. You are advised to compress the broadcast variables. using **spark.io.compression.codec**.	true
spark.reducer.maxSizeInFlight	Specifies the maximum output size of the Map task that fetches data from each Reduce task, in MB. Each output requires a buffer, which is the fixed memory overhead of each Reduce task. Therefore, keep the value small unless there is a large amount of memory. You can also set this parameter by setting **spark.reducer.maxMbInFlight**.	48MB

Driver Configuration

Spark driver can be considered as the client of Spark applications. All code parsing is completed in this process. Therefore, the parameters of this process are especially important. The following describes how to configure parameters for Spark driver.

- **JavaOptions**: parameter following **-D** in the Java command, which can be obtained by **System.getProperty**
- **ClassPath**: path for loading the Java classes and Native library
- **Java Memory and Cores**: memory and CPU usage of the Java process
- **Spark Configuration**: Spark internal parameter, which is irrelevant to the Java process

Table 12-356 Parameter description

Parameter	Description	Default Value
spark.driver.extraJavaOptions	Indicates a series of extra JVM options passed to the driver, for example, GC setting and logging. Note: In client mode, this configuration cannot be set directly in the application using SparkConf because the driver JVM has been started. You can use **--driver-java-options** or the default property file to set the parameter.	For details, see **Configuring Parameters Rapidly**.
spark.driver.extraClassPath	Indicates the extra class path entries attached to the class path of the driver. Note: In client mode, this configuration cannot be set directly in the application using SparkConf because the driver JVM has been started. You can use **--driver-java-options** or the default property file to set the parameter.	For details, see **Configuring Parameters Rapidly**.
spark.driver.userClassPathFirst	(Trial) Indicates whether to allow JAR files added by users to take precedence over Spark JAR files when classes are loaded in the driver. This feature can be used to mitigate conflicts between Spark dependencies and user dependencies. This feature is in the trial phase and is used only in cluster mode.	false
spark.driver.extraLibraryPath	Sets a special library path for starting the driver JVM. Note: In client mode, this configuration cannot be set directly in the application using SparkConf because the driver JVM has been started. You can use **--driver-java-options** or the default property file to set the parameter.	• JDBCServer2x: ${SPARK_INSTALL_HOME}/spark/native • SparkResource2x: ${DATA_NODE_INSTALL_HOME}/hadoop/lib/native

Parameter	Description	Default Value
spark.driver.cores	Specifies the number of cores used by the driver process. This parameter is available only in cluster mode.	1
spark.driver.memory	Indicates the memory used by the driver process, that is, the memory used by the SparkContext initialization process (for example, 512 MB and 2 GB). Note: In client mode, this configuration cannot be set directly in the application using SparkConf because the driver JVM has been started. You can use **--driver-java-options** or the default property file to set the parameter.	4G
spark.driver.maxResultSize	Indicates the total size of serialization results of all partitions for each Spark action operation (for example, collect). The value must be at least 1 MB. If this parameter is set to **0**, the size is not limited. If the total amount exceeds this limit, the task will be aborted. If the value is too large, the memory of the driver may be insufficient (depending on the object memory overhead of **spark.driver.memory** and JVM). Set a proper limit to ensure sufficient memory for the driver.	1G
spark.driver.host	Specifies the host name or IP address for the driver to listen on, which is used for the driver to communicate with the executor.	(local hostname)
spark.driver.port	Specifies the port for the driver to listen on, which is used for the driver to communicate with the executor.	(random)

ExecutorLauncher Configuration

ExecutorLauncher exists only in Yarn-client mode. In Yarn-client mode, ExecutorLauncher and the driver are not in the same process. Therefore, you need to configure parameters for ExecutorLauncher.

Table 12-357 Parameter description

Parameter	Description	Default Value
spark.yarn.am.extraJavaOptions	Indicates a string of extra JVM options to pass to the YARN ApplicationMaster in client mode. Use **spark.driver.extraJavaOptions** in cluster mode.	For details, see **Configuring Parameters Rapidly**.
spark.yarn.am.memory	Indicates the amount of memory to use for the YARN ApplicationMaster in client mode, in the same format as JVM memory strings (for example, 512 MB or 2 GB). In cluster mode, use **spark.driver.memory** instead.	1G
spark.yarn.am.memoryOverhead	This parameter is the same as **spark.yarn.driver.memoryOverhead**. However, this parameter applies only to ApplicationMaster in client mode.	-
spark.yarn.am.cores	Indicates the number of cores to use for the YARN ApplicationMaster in client mode. Use **spark.driver.cores** in cluster mode.	1

Executor Configuration

An executor is a Java process. However, unlike the driver and ApplicationMaster, an executor can have multiple processes. Spark supports only same configurations. That is, the process parameters of all executors must be the same.

Table 12-358 Parameter description

Parameter	Description	Default Value
spark.executor.extraJavaOptions	Indicates extra JVM option passed to the executor, for example, GC setting and logging. Do not set Spark attributes or heap size using this option. Instead, set Spark attributes using the SparkConf object or the **spark-defaults.conf** file specified when the spark-submit script is called. Set heap size using **spark.executor.memory**.	For details, see **Configuring Parameters Rapidly**.

Parameter	Description	Default Value
spark.executor.extraClassPath	Indicates the extra classpath attached to the executor classpath. This parameter ensures compatibility with historical versions of Spark. Generally, you do not need to set this parameter.	-
spark.executor.extraLibraryPath	Sets the special library path used when the executor JVM is started.	For details, see **Configuring Parameters Rapidly**.
spark.executor.userClassPathFirst	(Trial) Same function as **spark.driver.userClassPathFirst**. However, this parameter applies to executor instances.	false
spark.executor.memory	Indicates the memory size used by each executor process. Its character sting is in the same format as the JVM memory (example: 512 MB or 2 GB).	4G
spark.executorEnv.[EnvironmentVariableName]	Adds the environment variable specified by **EnvironmentVariableName** to the executor process. You can specify multiple environment variables.	-
spark.executor.logs.rolling.maxRetainedFiles	Sets the number of latest log files to be retained by the system during rolling. The old log files are deleted. This function is disabled by default.	-
spark.executor.logs.rolling.size.maxBytes	Sets the maximum size of the executor log file for rolling. This function is disabled by default. The value is in bytes. To automatically clear old logs, see **spark.executor.logs.rolling.maxRetainedFiles**.	-
spark.executor.logs.rolling.strategy	Sets the executor log rolling policy. Rolling is disabled by default. The value can be **time** (time-based rolling) or **size** (size-based rolling). If this parameter is set to **time**, the value of the **spark.executor.logs.rolling.time.interval** attribute is used as the log rolling interval. If this parameter is set to **size**, **spark.executor.logs.rolling.size.maxBytes** is used to set the maximum size of the file for rolling.	-

Parameter	Description	Default Value
spark.executor.logs.rolling.time.interval	Sets the time interval for executor log rolling. This function is disabled by default. The value can be **daily**, **hourly**, **minutely**, or any number of seconds. To automatically clear old logs, see **spark.executor.logs.rolling.maxRetainedFiles**.	daily

WebUI

The Web UI displays the running process and status of the Spark application.

Table 12-359 Parameter description

Parameter	Description	Default Value
spark.ui.killEnabled	Allows stages and jobs to be stopped on the web UI. **NOTE** For security purposes, the default value of this parameter is set to **false** to prevent misoperations. To enable this function, set this parameter to **true** in the **spark-defaults.conf** configuration file. Exercise caution when performing this operation.	true
spark.ui.port	Specifies the port for your application's dashboard, which displays memory and workload data.	• JDBCServer2x: **4040** • SparkResource2x: 0 • IndexServer2x: 22901
spark.ui.retainedJobs	Specifies the number of jobs recorded by the Spark UI and status API before GC.	1000
spark.ui.retainedStages	Specifies the number of stages recorded by the Spark UI and status API before GC.	1000

HistoryServer

A History Server reads the **EventLog** file in the file system and displays the running status of the Spark application.

Table 12-360 Parameter description

Parameter	Description	Default Value
spark.history.fs.logDirectory	Specifies the log directory of a History Server.	-
spark.history.ui.port	Specifies the port for JobHistory listening to connection.	18080
spark.history.fs.updateInterval	Specifies the update interval of the information displayed on a History Server, in seconds. Each update checks for changes made to the event logs in the persistent store.	10s
spark.history.fs.update.interval.seconds	Specifies the interval for checking the update of each event log. This parameter has the same function as **spark.history.fs.updateInterval**. **spark.history.fs.updateInterval** is recommended.	10s
spark.history.updateInterval	This parameter has the same function as **spark.history.fs.update.interval.seconds** and **spark.history.fs.updateInterval**. **spark.history.fs.updateInterval** is recommended.	10s

History Server UI Timeout and Maximum Number of Access Times

Table 12-361 Parameter description

Parameter	Description	Default Value
spark.session.maxAge	Specifies the session timeout interval, in seconds. This parameter applies only to the security mode. This parameter cannot be set in normal mode.	600
spark.connection.maxRequest	Specifies the maximum number of concurrent client access requests to JobHistory.	5000

EventLog

During the running of Spark applications, the running status is written into the file system in JSON format in real time for the History Server service to read and reproduce the application running status.

Table 12-362 Parameter description

Parameter	Description	Default Value
spark.eventLog.enabled	Indicates whether to log Spark events, which are used to reconstruct the web UI after the application execution is complete.	true
spark.eventLog.dir	Indicates the directory for logging Spark events if **spark.eventLog.enabled** is set to **true**. In this directory, Spark creates a subdirectory for each application and logs events of the application in the subdirectory. You can also set a unified address similar to the HDFS directory so that the History Server can read historical files.	hdfs://hacluster/spark2xJobHistory2x
spark.eventLog.compress	Indicates whether to compress logged events when **spark.eventLog.enabled** is set to **true**.	false

Periodic Clearing of Event Logs

Event logs on JobHistory increases with submitted tasks. Too many event log files exist as the number of submitted tasks increases. Spark provides the function for periodically clearing event logs. You can enable this function and set the clearing interval using related parameters.

Table 12-363 Parameter description

Parameter	Description	Default Value
spark.history.fs.cleaner.enabled	Indicates whether to enable the clearing function.	true
spark.history.fs.cleaner.interval	Indicates the check interval of the clearing function.	1d
spark.history.fs.cleaner.maxAge	Indicates the maximum duration for storing logs.	4d

Kryo

Kryo is a highly efficient Java serialization framework, which is integrated into Spark by default. Almost all Spark performance tuning requires the process of converting the default serializer of Spark into a Kryo serializer. Kryo serialization

supports only serialization at the Spark data layer. To configure Kryo serialization, set **spark.serializer** to **org.apache.spark.serializer.KryoSerializer** and configure the following parameters to optimize Kryo serialization performance:

Table 12-364 Parameter description

Parameter	Description	Default Value
spark.kryo.classesToRegister	Specifies the name of the class that needs to be registered with Kryo when Kryo serialization is used. Multiple classes are separated by commas (,).	-
spark.kryo.referenceTracking	Indicates whether to trace the references to the same object when Kryo is used to serialize data. This function is applicable to the scenario where the object graph has circular references or the same object has multiple copies. Otherwise, you can disable this function to improve performance.	true
spark.kryo.registrationRequired	Indicates whether Kryo is used to register an object. When this parameter is set to **true**, an exception is thrown if an object that is not registered with Kryo is serialized. When it is set to **false** (default value), Kryo writes unregistered class names to the serialized object. This operation causes a large amount of performance overhead. Therefore, you need to enable this option before deleting a class from the registration queue.	false
spark.kryo.registrator	If Kryo serialization is used, use Kryo to register the class with the custom class. Use this property if you need to register a class in a custom way, such as specifying a custom field serializer. Otherwise, use **spark.kryo.classesToRegister**, which is simpler. Set this parameter to a class that extends KryoRegistrator.	-
spark.kryoserializer.buffer.max	Specifies the maximum size of the Kryo serialization buffer, in MB. The value must be greater than the object that attempts to be serialized. If the error "buffer limit exceeded" occurs in Kryo, increase the value of this parameter. You can also set this parameter by setting **spark.kryoserializer.buffer.max**.	64MB

Parameter	Description	Default Value
spark.kryoserializer.buffer	Specifies the initial size of the Kryo serialization buffer, in MB. Each core of each worker has a buffer. If necessary, the buffer size will be increased to the value of **spark.kryoserializer.buffer.max**. You can also set this parameter by setting **spark.kryoserializer.buffer**.	64KB

Broadcast

Broadcast is used to transmit data blocks between Spark processes. In Spark, broadcast can be used for JAR packages, files, closures, and returned results. Broadcast supports two modes: Torrent and HTTP. The Torrent mode divides data into small fragments and distributes them to clusters. Data can be obtained remotely if necessary. The HTTP mode saves files to the local disk and transfers the entire files to the remote end through HTTP if necessary. The former is more stable than the latter. Therefore, Torrent is the default broadcast mode.

Table 12-365 Parameter description

Parameter	Description	Default Value
spark.broadcast.factory	Indicates the broadcast mode.	org.apache.spark.broadcast.TorrentBroadcastFactory
spark.broadcast.blockSize	Indicates the block size of **TorrentBroadcastFactory**. If the value is too large, the concurrency during broadcast is reduced (the speed is slow). If the value is too small, BlockManager performance may be affected.	4096
spark.broadcast.compress	Indicates whether to compress broadcast variables before sending them. You are advised to compress the broadcast variables.	true

Storage

Spark features in-memory computing. Spark Storage is used to manage memory resources. Storage stores data blocks generated during RDD caching. The heap memory in the JVM acts as a whole. Therefore, **Storage Memory Size** is an important concept during Spark Storage management.

Table 12-366 Parameter description

Parameter	Description	Default Value
spark.storage.memoryMapThreshold	Specifies the block size. If the size of a block exceeds the value of this parameter, Spark performs memory mapping for the disk file. This prevents Spark from mapping too small blocks during memory mapping. Generally, memory mapping for blocks whose page size is close to or less than that of the operating system has high overhead.	2m

PORT

Table 12-367 Parameter description

Parameter	Description	Default Value
spark.ui.port	Specifies the port for your application's dashboard, which displays memory and workload data.	• JDBCServer2x: **4040** • SparkResource2x: 0
spark.blockManager.port	Specifies all ports listened by BlockManager. These ports are on both the driver and executor.	**Range of Random Ports**
spark.driver.port	Specifies the port for the driver to listen on, which is used for the driver to communicate with the executor.	**Range of Random Ports**

Range of Random Ports

All random ports must be within a certain range.

Table 12-368 Parameter description

Parameter	Description	Default Value
spark.random.port.min	Sets the minimum random port.	22600
spark.random.port.max	Sets the maximum random port.	22899

TIMEOUT

By default, computation tasks that can well process medium-scale data are configured in Spark. However, if the data volume is too large, the tasks may fail due to timeout. In the scenario with a large amount of data, the timeout parameter in Spark needs to be assigned a larger value.

Table 12-369 Parameter description

Parameter	Description	Default Value
spark.files.fetchTimeout	Specifies the communication timeout (in seconds) when fetching files added using **SparkContext.addFile()** of the driver.	60s
spark.network.timeout	Specifies the default timeout for all network interactions, in seconds. You can use this parameter to replace **spark.core.connection.ack.wait.timeout**, **spark.akka.timeout**, **spark.storage.blockManagerSlaveTimeoutMs**, or **spark.shuffle.io.connectionTimeout**.	360s
spark.core.connection.ack.wait.timeout	Specifies the timeout for a connection to wait for a response, in seconds. To avoid long-time waiting caused by GC, you can set this parameter to a larger value.	60

Encryption

Spark supports SSL for Akka and HTTP (for the broadcast and file server) protocols, but does not support SSL for the web UI and block transfer service.

SSL must be configured on each node and configured for each component involved in communication using a particular protocol.

Table 12-370 Parameter description

Parameter	Description	Default Value
spark.ssl.enabled	Indicates whether to enable SSL connections for all supported protocols. All SSL settings similar to **spark.ssl.*xxx*** indicate the global configuration of all supported protocols. To override the global configuration of a particular protocol, you must override the property in the namespace specified by the protocol. Use **spark.ssl.YYY.XXX** to overwrite the global configuration of the particular protocol specified by **YYY**. **YYY** can be either **akka** for Akka-based connections or **fs** for the broadcast and file server.	false
spark.ssl.enabledAlgorithms	Indicates the comma-separated list of passwords. The specified passwords must be supported by the JVM.	-
spark.ssl.keyPassword	Specifies the password of a private key in the keystore.	-
spark.ssl.keyStore	Specifies the path of the keystore file. The path can be absolute or relative to the directory where the component is started.	-
spark.ssl.keyStorePassword	Specifies the password of the keystore.	-
spark.ssl.protocol	Specifies the protocol name. This protocol must be supported by the JVM. The reference list of protocols is available on this page.	-
spark.ssl.trustStore	Specifies the path of the truststore file. The path can be absolute or relative to the directory where the component is started.	-
spark.ssl.trustStorePassword	Specifies the password of the truststore.	-

Security

Spark supports shared key-based authentication. You can use **spark.authenticate** to configure authentication. This parameter controls whether the Spark communication protocol uses the shared key for authentication. This authentication is a basic handshake that ensures that both sides have the same shared key and are allowed to communicate. If the shared keys are different, the communication is not allowed. You can create shared keys as follows:

- For Spark on Yarn deployments, set **spark.authenticate** to **true**. Then, shared keys are automatically generated and distributed. Each application exclusively occupies a shared key.

- For other types of Spark deployments, configure Spark parameter **spark.authenticate.secret** on each node. All masters, workers, and applications use this key.

Table 12-371 Parameter description

Parameter	Description	Default Value
spark.acls.enable	Indicates whether to enable Spark ACLs. If Spark ACLs are enabled, the system checks whether the user has the permission to access and modify jobs. Note that this requires the user to be identifiable. If the user is identified as invalid, the check will not be performed. Filters can be used to verify and set users on the UI.	true
spark.admin.acls	Specifies the comma-separated list of users/administrators that have the permissions to view and modify all Spark jobs. This list can be used if you are running on a shared cluster and working with the help of an administrator or developer.	admin
spark.authenticate	Indicates whether Spark authenticates its internal connections. If the application is not running on Yarn, see **spark.authenticate.secret**.	true
spark.authenticate.secret	Sets the key for authentication between Spark components. This parameter must be set if Spark does not run on Yarn and authentication is disabled.	-
spark.modify.acls	Specifies the comma-separated list of users who have the permission to modify Spark jobs. By default, only users who have enabled Spark jobs have the permission to modify the list (for example, delete the list).	-
spark.ui.view.acls	Specifies the comma-separated list of users who have the permission to access the Spark web UI. By default, only users who have enabled Spark jobs have the access permission.	-

Enabling the Authentication Mechanism Between Spark Processes

Spark processes support shared key-based authentication. You can configure **spark.authenticate** to control whether Spark performs authentication during communication. In this authentication mode, the two communication parties share the same key only using simple handshakes.

Configure the following parameters in the **spark-defaults.conf** file on the Spark client.

Table 12-372 Parameter description

Parameter	Description	Default Value
spark.authenticate	For Spark on Yarn deployments, set this parameter to **true**. Then, keys are automatically generated and distributed, and each application uses a unique key.	true

Compression

Data compression is policy that optimizes memory usage at the expense of CPU. Therefore, when the Spark memory is severely insufficient (this issue is common due to the characteristics of in-memory computing), data compression can greatly improve performance. Spark supports three types of compression algorithm: Snappy, LZ4, and LZF. Snappy is the default compression algorithm and invokes the native method to compress and decompress data. In Yarn mode, pay attention to the impact of non-heap memory on the container process.

Table 12-373 Parameter description

Parameter	Description	Default Value
spark.io.compression.codec	Indicates the codec for compressing internal data, such as RDD partitions, broadcast variables, and shuffle output. By default, Spark supports three types of compression algorithm: LZ4, LZF, and Snappy. You can specify algorithms using fully qualified class names, such as **org.apache.spark.io.LZ4CompressionCodec**, **org.apache.spark.io.LZFCompressionCodec**, and **org.apache.spark.io.SnappyCompression-Codec**.	lz4
spark.io.compression.lz4.block.size	Indicates the block size (bytes) used in LZ4 compression when the LZ4 compression algorithm is used. When LZ4 is used, reducing the block size also reduces the shuffle memory usage.	32768
spark.io.compression.snappy.block.size	Indicates the block size (bytes) used in Snappy compression when the Snappy compression algorithm is used. When Snappy is used, reducing the block size also reduces the shuffle memory usage.	32768
spark.shuffle.compress	Indicates whether to compress the output files of a Map task. You are advised to compress the broadcast variables. using **spark.io.compression.codec**.	true

Parameter	Description	Default Value
spark.shuffle.spill.compress	Indicates whether to compress the data overflowed during shuffle using **spark.io.compression.codec**.	true
spark.eventLog.compress	Indicates whether to compress logged events when **spark.eventLog.enabled** is set to **true**.	false
spark.broadcast.compress	Indicates whether to compress broadcast variables before sending them. You are advised to compress the broadcast variables.	true
spark.rdd.compress	Indicates whether to compress serialized RDD partitions (for example, the **StorageLevel.MEMORY_ONLY_SER** partition). Substantial space can be saved at the cost of some extra CPU time.	false

Reducing the Probability of Abnormal Client Application Operations When Resources Are Insufficient

When resources are insufficient, ApplicationMaster tasks must wait and will not be processed until enough resources are available for use. If the actual waiting time exceeds the configured waiting time, the ApplicationMaster tasks will be deleted. Adjust the following parameters to reduce the probability of abnormal client application operation.

Configure the following parameters in the **spark-defaults.conf** file on the client.

Table 12-374 Parameter description

Parameter	Description	Default Value
spark.yarn.applicationMaster.waitTries	Specifies the number of the times that ApplicationMaster waits for Spark master, which is also the times that ApplicationMaster waits for SparkContext initialization. Enlarge this parameter value to prevent ApplicationMaster tasks from being deleted and reduce the probability of abnormal client application operations.	10
spark.yarn.am.memory	Specifies the ApplicationMaster memory. Enlarge this parameter value to prevent ApplicationMaster tasks from being deleted by ResourceManager due to insufficient memory and reduce the probability of abnormal client application operations.	1G

12.23.2.4 Spark on HBase Overview and Basic Applications

Scenario

Spark on HBase allows users to query HBase tables in Spark SQL and to store data for HBase tables by using the Beeline tool. You can use HBase APIs to create, read data from, and insert data into tables.

Procedure

Step 1 Log in to Manager and choose **Cluster** > *Name of the desired cluster* > **Cluster Properties** to check whether the cluster is in security mode.

- If yes, go to **Step 2**.
- If no, go to **Step 5**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Service** > **Spark2x** > **Configuration** > **All Configurations** > **JDBCServer2x** > **Default**, and modify the following parameter.

Table 12-375 Parameter list 1

Parameter	Default Value	Changed To
spark.yarn.security.credentials.hbase.enabled	false	true

NOTE

To ensure that Spark2x can access HBase for a long time, do not modify the following parameters of the HBase and HDFS services:

- dfs.namenode.delegation.token.renew-interval
- dfs.namenode.delegation.token.max-lifetime
- hbase.auth.key.update.interval
- hbase.auth.token.max.lifetime (The value is fixed to **604800000** ms, that is, 7 days.)

If the preceding parameter configuration must be modified based on service requirements, ensure that the value of the HDFS parameter **dfs.namenode.delegation.token.renew-interval** is not greater than the values of the HBase parameters **hbase.auth.key.update.interval**, **hbase.auth.token.max.lifetime**, and **dfs.namenode.delegation.token.max-lifetime**.

Step 3 Choose **SparkResource2x** > **Default** and modify the following parameters.

Table 12-376 Parameter list 2

Parameter	Default Value	Changed To
spark.yarn.security.credentials.hbase.enabled	false	true

Step 4 Restart the Spark2x service for the configuration to take effect.

NOTE

To use the Spark on HBase function on the Spark2x client, you need to download and install the Spark2x client again.

Step 5 On the Spark2x client, use the spark-sql or spark-beeline connection to query tables created by Hive on HBase. You can create an HBase table by running SQL commands or create an external table to associate the HBase table. Before creating tables, ensure that HBase tables exist in HBase. The HBase table **table1** is used as an example.

1. Run the following commands to create the HBase table using the Beeline tool:

   create table *hbaseTable*

   (

   id string,

   name string,

   age int

)

   using org.apache.spark.sql.hbase.HBaseSource

   options(

   hbaseTableName *"table1"***,**

   keyCols *"id"***,**

   colsMapping "

   name=cf1.cq1,

   age=cf1.cq2

   ");

   NOTE

   - **hbaseTable**: name of the created Spark table
   - **id string,name string**, **age int**: field name and field type of the Spark table
   - **table1**: name of the HBase table
   - *id*: row key column name of the HBase table
   - *name=cf1.cq1*, *age=cf1.cq2*: mapping between columns in the Spark table and columns in the HBase table. The **name** column of the Spark table maps the **cq1** column in the **cf1** column family of the HBase table, and the **age** column of the Spark table maps the **cq2** column in the **cf1** column family of the HBase table.

2. Run the following command to import data to the HBase table using a CSV file:

   hbase org.apache.hadoop.hbase.mapreduce.ImportTsv -Dimporttsv.separator="," -Dimporttsv.columns=HBASE_ROW_KEY,*cf1:cq1,cf1:cq2,cf1:cq3,cf1:cq4,cf1:cq5 table1 /hperson*

   Where **table1** indicates the name of the HBase table, and **/hperson** indicates the path where the CSV file is stored.

3. Run the following command to query data in spark-sql or spark-beeline, where *hbaseTable* is the corresponding Spark table name: The command is as follows:

select * from *hbaseTable;*

----End

12.23.2.5 Spark on HBase V2 Overview and Basic Applications

Scenario

Spark on HBase V2 allows users to query HBase tables in Spark SQL and to store data for HBase tables by using the Beeline tool. You can use HBase APIs to create, read data from, and insert data into tables.

Procedure

Step 1 Log in to Manager and choose **Cluster** > *Name of the desired cluster* > **Cluster Properties** to check whether the cluster is in security mode.

- If yes, go to **Step 2**.
- If no, go to **Step 5**.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Service** > **Spark2x** > **Configuration** > **All Configurations** > **JDBCServer2x** > **Default**, and modify the following parameter.

Table 12-377 Parameter list 1

Parameter	Default Value	Changed To
spark.yarn.security.credentials.hbase.enabled	false	true

NOTE

To ensure that Spark2x can access HBase for a long time, do not modify the following parameters of the HBase and HDFS services:

- dfs.namenode.delegation.token.renew-interval
- dfs.namenode.delegation.token.max-lifetime
- hbase.auth.key.update.interval
- hbase.auth.token.max.lifetime (The value is fixed to **604800000** ms, that is, 7 days.)

If the preceding parameter configuration must be modified based on service requirements, ensure that the value of the HDFS parameter **dfs.namenode.delegation.token.renew-interval** is not greater than the values of the HBase parameters **hbase.auth.key.update.interval**, **hbase.auth.token.max.lifetime**, and **dfs.namenode.delegation.token.max-lifetime**.

Step 3 Choose **SparkResource2x** > **Default** and modify the following parameters.

Table 12-378 Parameter list 2

Parameter	Default Value	Changed To
spark.yarn.security.credentials.hbase.enabled	false	true

Step 4 Restart the Spark2x service for the configuration to take effect.

NOTE

If you need to use the Spark on HBase function on the Spark2x client, download and install the Spark2x client again.

Step 5 On the Spark2x client, use the spark-sql or spark-beeline connection to query tables created by Hive on HBase. You can create an HBase table by running SQL commands or create an external table to associate the HBase table. For details, see the following description. The following uses the HBase table **table1** as an example.

1. Run the following commands to create a table using the spark-beeline tool:

   create table *hbaseTable*1
   (*id string, name string, age int***)**
   using org.apache.spark.sql.hbase.HBaseSourceV2
   options(
   hbaseTableName *"table2"***,**
   keyCols *"id"***,**
   colsMapping "*name=cf1.cq1,age=cf1.cq2***");**

   NOTE

   - **hbaseTable1**: name of the created Spark table
   - **id string,name string**, **age int**: field name and field type of the Spark table
   - **table2**: name of the HBase table
   - *id*: row key column name of the HBase table
   - *name=cf1.cq1*, *age=cf1.cq2*: mapping between columns in the Spark table and columns in the HBase table. The **name** column of the Spark table maps the **cq1** column in the **cf1** column family of the HBase table, and the **age** column of the Spark table maps the **cq2** column in the **cf1** column family of the HBase table.

2. Run the following command to import data to the HBase table using a CSV file:

   hbase org.apache.hadoop.hbase.mapreduce.ImportTsv -Dimporttsv.separator="," -Dimporttsv.columns=HBASE_ROW_KEY,*cf1:cq1,cf1:cq2,cf1:cq3,cf1:cq4,cf1:cq5 table2 /hperson*

   Where **table2** indicates the name of the HBase table, and **/hperson** indicates the path where the CSV file is stored.

3. Run the following command to query data in spark-sql or spark-beeline. *hbaseTable1* indicates the corresponding Spark table name.

   select * from *hbaseTable1;*

----End

12.23.2.6 SparkSQL Permission Management(Security Mode)

12.23.2.6.1 Spark SQL Permissions

SparkSQL Permissions

Similar to Hive, Spark SQL is a data warehouse framework built on Hadoop, providing storage of structured data like structured query language (SQL).

MRS supports users, user groups, and roles. Permission must be assigned to roles and then roles are bound to users or user groups. Users can obtain permissions only by binding a role or joining a group that is bound with a role.

NOTE

- If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Spark2x**.
- After Ranger authentication is enabled or disabled on Spark2x, you need to restart Spark2x and download the client again or update the client configuration file **spark/conf/spark-defaults.conf**.

  Enable Ranger authentication: **spark.ranger.plugin.authorization.enable=true**

  Disable Ranger authentication: **spark.ranger.plugin.authorization.enable=false**

Permission Management

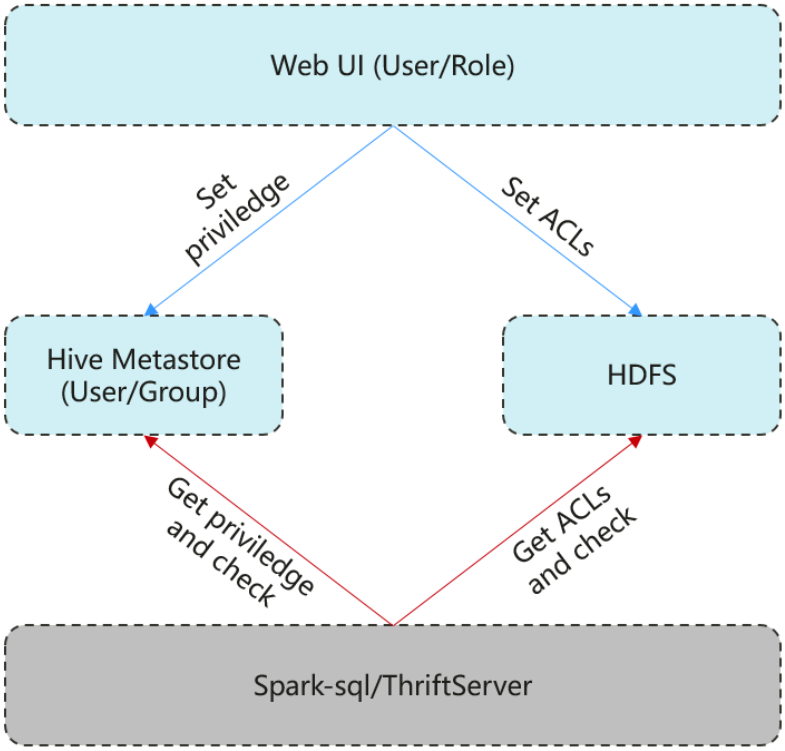
Spark SQL permission management indicates the permission system for managing and controlling users' operations on databases, to ensure that different users can operate databases separately and securely. A user can operate another user's tables and databases only with the corresponding permissions. Otherwise, operations will be rejected.

Spark SQL permission management integrates the functions of Hive management. The MetaStore service of Hive and the permission granting function on the page are required to enable Spark SQL permission management.

Figure 12-53 shows the basic architecture of SparkSQL permission management. This architecture includes two parts: granting permissions on the page, and obtaining and judging a service.

- Granting permissions on the page: Spark SQL only supports granting permissions on the page. On FusionInsight Manager, choose **System** > **Permission** to add or delete a user, user group, or a role, and to grant permissions or cancel permissions.
- Obtaining and judging a service: When the DDL and DML commands are received from a client, Spark SQL will obtain the client's permissions on database information from MetaStore, and check whether the required permissions are included. If the required permissions are included, continue the execution. If the required permissions are not included, reject the user's operations. After the MetaStore permissions are checked, ACL permission also needs to be checked on HDFS.

Figure 12-53 Spark SQL permission management architecture



Additionally, Spark SQL provides column and view permissions to meet requirements of different scenarios.

- Column permission

  Spark SQL permission control consists of metadata permission control and HDFS ACL permission control. When Hive MetaStore automatically synchronizes table permissions to the HDFS ACL, column-level permissions are not synchronized. In other words, a user with partial or all column-level permissions cannot access the entire HDFS file using the HDFS client.

  - In **spark-sql** mode, users with only column-level permissions cannot access HDFS files. Therefore, they cannot access the columns of the corresponding tables.
  - In Beeline/JDBCServer mode, permissions are assigned among users, for example, the permissions on the table created by user A are assigned to user B.

    - **hive.server2.enable.doAs**=**true** (configured in the **hive-site.xml** file on the Spark server)

      In this case, user B cannot query the information. You need to manually assign the read permission on the file in HDFS.

    - **hive.server2.enable.doAs**=false
      - Users A and B are connected by Beeline. User B can query the information.
      - User A creates a table using SQL statements, and user B can query the table in Beeline.

      However, information query is not supported in other scenarios, for example, user A uses Beeline to create a table and user B uses SQL

to query the table, or user A uses SQL to create a table and user B uses SQL to query the table. You need to manually assign the read permission on the file in HDFS.

NOTE

The **spark** user is an administrator in HDFS ACL permission control. The permission control of the Beeline client user depends only on the metadata permission on Spark.

- View permission

  View permission indicates the operation permission such as query and modification on the view of a table, regardless of the corresponding permission of a table. Namely, if you have the permission to query the view of a table, the permission to query the table is not mandatory. The view permission is applicable to the whole table but not to the columns.

  Restrictions of view and column permissions on SparkSQL are similar. The following uses the view permission as an example:

  - In spark-sql mode, if you have only the view permission but not the table permission and do not have the permission to read HDFS, you cannot access the table data stored in HDFS. That is, you cannot query the view of the table.
  - In Beeline/JDBCServer mode, permissions are assigned among users, for example, the permissions on the view created by user A are assigned to user B.

    - **hive.server2.enable.doAs**=**true** (configured in the **hive-site.xml** file on the Spark server)

      In this case, user B cannot query the information. You need to manually assign the read permission on the file in HDFS.

    - **hive.server2.enable.doAs**=false
      - Users A and B are connected by Beeline. User B can query the information.
      - User A creates a view using SQL statements, and user B can query the view in Beeline.

      However, information query is not supported in other scenarios. For example, user A uses Beeline to create a view but user B cannot use SQL to query the view, or user A uses SQL to create a view but user B cannot use SQL to query the view. You need to manually assign the read permission on the file in HDFS.

  Permission of operations on the view of a table is as follows:

  - To create a view, you must have the CREATE permission on the database and the SELECT and SELECT_of_GRANT permissions on the tables.
  - Creating and describing a view only entail the SELECT permission on the view. Querying views and tables at the same time entails the SELECT permission on other tables. For example, to perform **select * from v1 join t1**, you must have the SELECT permission on the **v1** view and **t1** table, even through the **v1** view depends on the **t1** table.

NOTE

In Beeline/JDBCServer mode, to query a view, you must have the SELECT permission on the tables. In spark-sql mode, to query a view, you must have the SELECT permission on the view and tables.

- Deleting and modifying a view entail the permission of owner on the view.

SparkSQL Permission Model

If you want to perform SQL operations using SparkSQL, you must be granted with permissions of SparkSQL databases and tables (include external tables and views). The complete permission model of SparkSQL consists of the meta data permission and HDFS file permission. Permissions required to use a database or a table is just one type of SparkSQL permission.

- Metadata permissions

  Metadata permissions are controlled at the metadata layer. Similar to traditional relational databases, SparkSQL databases involve the CREATE and SELECT permissions, and tables and columns involve the SELECT, INSERT, UPDATE, and DELETE permissions. SparkSQL also supports the permissions of **OWNERSHIP** and **ADMIN**.

- Data file permissions (that is, HDFS file permissions)

  SparkSQL database and table files are stored in HDFS. The created databases or tables are saved in the **/user/hive/warehouse** directory of HDFS by default. The system automatically creates subdirectories named after database names and database table names. To access a database or table, you must have the **Read**, **Write** and **Execute** permissions on the corresponding file in HDFS.

To perform various operations on SparkSQL databases or tables, you need to associate the metadata permission and HDFS file permission. For example, to query SparkSQL data tables, you need to associate the metadata permission **SELECT** and HDFS file permissions **Read** and **Execute**.

Using the management function of Manager GUI to manage the permissions of SparkSQL databases and tables, only requires the configuration of metadata permission, and the system will automatically associate and configure the HDFS file permission. In this way, operations on the interface are simplified, and the efficiency is improved.

Usage Scenarios and Related Permissions

Creating a database with SparkSQL service requires users to join in the hive group, without granting a role. Users have all permissions on the databases or tables created by themselves in Hive or HDFS. They can create tables, select, delete, insert, or update data, and grant permissions to other users to allow them to access the tables and corresponding HDFS directories and files.

A user can access the tables or database only with permissions. Users' permissions vary depending on different SparkSQL scenarios.

Table 12-379 SparkSQL scenarios

Typical Scenario	Required Permission
Using SparkSQL tables, columns, or databases	Permissions required in different scenarios are as follows: ● To create a table, the CREATE permission is required. ● To query data, the SELECT permission is required. ● To insert data, the INSERT permission is required.
Associating and using other components	In some scenarios, except the SparkSQL permission, other permissions may be also required. For example: Using Spark on HBase to query HBase data in SparkSQL requires HBase permissions.

In some special SparkSQL scenarios, other permissions must be configured separately.

Table 12-380 SparkSQL scenarios and required permissions

Scenario	Required Permission
Creating SparkSQL databases, tables, and external tables, or adding partitions to created Hive tables or external tables when data files specified by Hive users are saved to other HDFS directories except **/user/hive/warehouse**	● The directory must exist, the client user must be the owner of the directory, and the user must have the **Read**, **Write**, and **Execute** permissions on the directory. The user must have the **Read** and **Execute** permissions of all the upper-layer directories of the directory. ● If the Spark version is later than 2, the **Create** permission of the Hive database is required if you want to create a HBase table. However, in Spark 1.5, the **Create** permissions of both the Hive database and HBase namespace are required if you want to create a HBase table.

Scenario	Required Permission
Importing all the files or specified files in a specified directory to the table using load	• The data source is a Linux local disk, the specified directory exists, and the system user **omm** has read and execute permission of the directory and all its upper-layer directories. The specified file exists, and user **omm** has the **Read** permission on the file and has the **Read** and **Execute** permissions on all the upper-layer directories of the file. • The data source is HDFS, the specified directory exists, and the SparkSQL user is the owner of the directory and has the **Read**, **Write**, and **Execute** permissions on the directory and its subdirectories, and has the **Read** and **Execute** permissions on all its upper-layer directories. The specified file exists, and the SparkSQL user is the owner of the file and has the **Read**, **Write**, and **Execute** permissions on the file and has the **Read** and **Execute** permissions on all its upper-layer directories.
Creating or deleting functions or modifying any database	The **ADMIN** permission is required.
Performing operations on all databases and tables in Hive	The user must be added to the **supergroup** user group, and be assigned the **ADMIN** permission.
After assigning the **Insert** permission on some DataSource tables, assigning the **Write** permission on table directories in HDFS before performing the insert or analyze operation	When the **Insert** permission is assigned to the **spark datasource** table, if the table format is text, CSV, JSON, Parquet, or ORC, the permission on the table directory is not changed. After the **Insert** permission is assigned to the DataSource table of the preceding formats, you need to assign the **Write** permission to the table directories in HDFS separately so that users can perform the insert or analyze operation on the tables.

12.23.2.6.2 Creating a Spark SQL Role

Scenario

This section describes how to create and configure a SparkSQL role on Manager as the system administrator. The Spark SQL role can be configured with the administrator permission or the permission of performing operations on the table data.

Creating a database with Hive requires users to join in the **hive** group, without granting a role. Users have all permissions on the databases or tables created by themselves in Hive or HDFS. They can create tables, select, delete, insert, or update data, and grant permissions to other users to allow them to access the

tables and corresponding HDFS directories and files. The created databases or tables are saved in the **/user/hive/warehouse** directory of HDFS by default.

NOTE

- If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. For details, see **Adding a Ranger Access Permission Policy for Spark2x**.
- After Ranger authentication is enabled or disabled on Spark2x, you need to restart Spark2x and download the client again or update the client configuration file **spark/conf/spark-defaults.conf**.

  Enable Ranger authentication: **spark.ranger.plugin.authorization.enable=true**

  Disable Ranger authentication: **spark.ranger.plugin.authorization.enable=false**

Procedure

1. Log in to Manager, and choose **System > Permission > Role**.
2. Click **Create Role** and set a role name and enter description.
3. Set **Configure Resource Permission**. For details, see **Table 12-381**.
   - **Hive Admin Privilege**: Hive administrator permissions.
   - **Hive Read Write Privileges**: Hive data table management permission, which is the operation permission to set and manage the data of created tables.

     NOTE

     - Hive role management supports the administrator permission, and the permissions of accessing tables and views, without granting the database permission.
     - The permissions of the Hive administrator do not include the permission to manage HDFS.
     - If there are too many tables in the database or too many files in tables, the permission granting may last a while. For example, if a table contains 10,000 files, the permission granting lasts about 2 minutes.

Table 12-381 Setting a role

Task	Operation
Hive administrator permission	In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** and select **Hive Admin Privilege**. After being bound to the Hive administrator role, perform the following operations during each maintenance operation: 1. Log in to the node where the Spark2x client is installed as the client installation user. 2. Run the following command to configure environment variables: For example, if the Spark2x client installation directory is **/opt/client**, run **source /opt/client/bigdata_env**. **source /opt/client/Spark2x/component_env** 3. Run the following command to perform user authentication: ***kinit*** *Hive service user* 4. Run the following command to log in to the client tool: **/opt/client/Spark2x/spark/bin/beeline -u "jdbc:hive2://<zkNode1_IP>:<zkNode1_Port>,<zkNode2_IP>:<zkNode2_Port>,<zkNode3_IP>:<zkNode3_Port>/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;user.principal=spark2x/hadoop.***<*system domain name*>**@**<*system domain name*>**;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.***<*system domain name*>**@**<*system domain name*>**;"**

Task	Operation
	NOTE • **<zkNode1_IP>:<zkNode1_Port>,<zkNode2_IP>:<zkNode2_Port>,<zkNode3_IP>:<zkNode3_Port>** indicates the ZooKeeper URL, for example, 192.168.81.37:2181,192.168.195.232:2181,192.168.169.84:2181. • **sparkthriftserver** indicates a ZooKeeper directory, from which a random TriftServer or ProxyThriftServer is connected by the client. • You can log in to Manager, choose **System** > **Permission** > **Domain and Mutual Trust**, and view the value of **Local Domain**, which is the current system domain name. **spark2x/hadoop.**<*System domain name*> is the username. All letters in the system domain name contained in the username are lowercase letters. For example, **Local Domain** is set to **9427068F-6EFA-4833-B43E-60CB641E5B6C.COM**, and the username is **spark2x/hadoo.9427068f-6efa-4833-b43e-60cb641e5b6c.com**. 5. Run the following command to update the administrator permissions: **set role admin;**
Setting the permission to query a table of another user in the default database	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**. 2. Click the name of the specified database in the database list. Tables in the database are displayed. 3. In the **Permission** column of the specified table, select **SELECT**.
Setting the permission to import data to a table of another user in the default database	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**. 2. Click the name of the specified database in the database list. Tables in the database are displayed. 3. In the **Permission** column of the specified table, select **DELETE** and **INSERT**.

4. Click **OK**.

12.23.2.6.3 Configuring Permissions for SparkSQL Tables, Columns, and Databases

Scenario

You can configure related permissions if you need to access tables or databases created by other users. SparkSQL supports column-based permission control. If a

user needs to access some columns in tables created by other users, the user must be granted the permission for columns. The following describes how to grant table, column, and database permissions to users by using the role management function of Manager.

Procedure

The operations for granting permissions on SparkSQL tables, columns, and databases are the same as those for Hive. For details, see **Permission Management**.

NOTE

- Any permission for a table in the database is automatically associated with the HDFS permission for the database directory to facilitate permission management. When any permission for a table is canceled, the system does not automatically cancel the HDFS permission for the database directory to ensure performance. In this case, users can only log in to the database and view table names.
- When the query permission on a database is added to or deleted from a role, the query permission on tables in the database is automatically added to or deleted from the role. This mechanism is inherited from Hive.
- In Spark, the column name of the struct data type cannot contain special characters, that is, characters other than letters, digits, and underscores (_). If the column name of the struct data type contains special characters, the column cannot be displayed on the FusionInsight Manager console when you grant permissions to roles on the role page.

Concepts

SparkSQL statements are processed in SparkSQL. **Table 12-382** describes the permission requirements.

Table 12-382 Scenarios of using SparkSQL tables, columns, or databases

Scenario	Required Permission
CREATE TABLE	**CREATE**, RWX+ownership (for creating external tables - the location) **NOTE** When creating datasource tables in a specified file path, the RWX and ownership permission on the file next to the path is required.
DROP TABLE	**Ownership** (of table)
DROP TABLE PROPERTIES	**Ownership**
DESCRIBE TABLE	**Select**
SHOW PARTITIONS	**Select**
ALTER TABLE LOCATION	**Ownership**, RWX+ownership (for new location)
ALTER PARTITION LOCATION	**Ownership**, RWX+ownership (for new partition location)
ALTER TABLE ADD PARTITION	**Insert**, RWX and ownership (for partition location)

Scenario	Required Permission
ALTER TABLE DROP PARTITION	**Delete**
ALTER TABLE(all of them except the ones above)	**Update**, **Ownership**
TRUNCATE TABLE	**Ownership**
CREATE VIEW	**Select**, **Grant Of Select**, **CREATE**
ALTER VIEW PROPERTIES	**Ownership**
ALTER VIEW RENAME	**Ownership**
ALTER VIEW ADD PARTS	**Ownership**
ALTER VIEW AS	**Ownership**
ALTER VIEW DROPPARTS	**Ownership**
ANALYZE TABLE	**Search**, **Insert**
SHOW COLUMNS	**Select**
SHOW TABLE PROPERTIES	**Select**
CREATE TABLE AS SELECT	**Select**, **CREATE**
SELECT	**Select** **NOTE** The same as tables, you need to have the **Select** permission on a view when performing a SELECT operation on the view.
INSERT	**Insert**, **Delete (for overwrite)**
LOAD	**Insert**, **Delete**, RWX+ownership(input location)
SHOW CREATE TABLE	**Select**, **Grant Of Select**
CREATE FUNCTION	**ADMIN**
DROP FUNCTION	**ADMIN**
DESC FUNCTION	-
SHOW FUNCTIONS	-
MSCK (metastore check)	**Ownership**
ALTER DATABASE	**ADMIN**
CREATE DATABASE	-
SHOW DATABASES	-
EXPLAIN	**Select**
DROP DATABASE	**Ownership**

Scenario	Required Permission
DESC DATABASE	-
CACHE TABLE	**Select**
UNCACHE TABLE	**Select**
CLEAR CACHE TABLE	**ADMIN**
REFRESH TABLE	**Select**
ADD FILE	**ADMIN**
ADD JAR	**ADMIN**
HEALTHCHECK	-

12.23.2.6.4 Configuring Permissions for SparkSQL to Use Other Components

Scenario

SparkSQL may need to be associated with other components. For example, Spark on HBase requires HBase permissions. The following describes how to associate SparkSQL with HBase.

Prerequisites

- The Spark client has been installed. For example, the installation directory is **/opt/client**.
- You have obtained a user account with the administrator permissions, such as **admin**.

Procedure

- **Spark on HBase authorization**

  After the permissions are assigned, you can use statements that are similar to SQL statements to access HBase tables from SparkSQL. The following uses the procedure for assigning a user the permissions to query HBase tables as an example.

  NOTE

  Set **spark.yarn.security.credentials.hbase.enabled** to **true**.

  a. On Manager, create a role, for example, **hive_hbase_create**, and grant the permission to create HBase tables to the role.

  In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global**. Select **create** of the namespace **default**, and click **OK**.

NOTE

In this example, the created table is saved in the default database of Hive and has the CREATE permission of the default database. If you save the table to a Hive database other than **default**, perform the following operations:

In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**, select **CREATE** for the desired database, and click **OK**.

b. On Manager, create a role, for example, **hive_hbase_submit**, and grant the permission to submit tasks to the Yarn queue.

In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Yarn** > **Scheduling Queue** > **root**. Select **Submit** of **default**, and click **OK**.

c. On Manager, create a human-machine user, for example, **hbase_creates_user**, add the user to the **hive** group, and bind the **hive_hbase_create** and **hive_hbase_submit** roles to create SparkSQL and HBase tables.

d. Log in to the node where the client is installed as the client installation user.

e. Run the following command to configure environment variables:

source /opt/client/bigdata_env

source /opt/client/Spark2x/component_env

f. Run the following command to authenticate the user:

kinit hbase_creates_user

g. Run the following commands to enter the shell environment on the Spark JDBCServer client:

/opt/client/Spark2x/spark/bin/beeline -u "jdbc:hive2://<zkNode1_IP>:<zkNode1_Port>,<zkNode2_IP>:<zkNode2_Port>,<zkNode3_IP>:<zkNode3_Port>/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;user.principal=spark2x/hadoop.*<system domain name>***@***<system domain name>***;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.***<system domain name>***@***<system domain name>***;"**

h. Run the following command to create a table in SparkSQL and HBase, for example, create the **hbaseTable** table:

create table hbaseTable (id string, name string, age int) using org.apache.spark.sql.hbase.HBaseSource options (hbaseTableName "table1", keyCols "id", colsMapping = ", name=cf1.cq1, age=cf1.cq2");

The created SparkSQL table and the HBase table are stored in the Hive database **default** and the HBase namespace **default**, respectively.

i. On Manager, create a role, for example, **hive_hbase_select**, and grant the role the permission to query SparkSQL on HBase table **hbaseTable** and HBase table **hbaseTable**.

- In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global** > **default**. Select **read** for the **hbaseTable** table, and click **OK** to grant the table query permission to the HBase role.

- Edit the role. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **HBase** > **HBase Scope** > **global** > **hbase**. Select **Execute** for **hbase:meta**, and click **OK**.
- Edit the role. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges** > **default**. Select **SELECT** for the **hbaseTable** table, and click **OK**.

j. On Manager, create a human-machine user, for example, **hbase_select_user**, add the user to the **hive** group, and bind the **hive_hbase_select** role to the user for querying SparkSQL and HBase tables.

k. Run the following command to configure environment variables:

source /opt/client/bigdata_env

source /opt/client/Spark2x/component_env

l. Run the following command to authenticate users:

kinit hbase_select_user

m. Run the following commands to enter the shell environment on the Spark JDBCServer client:

/opt/client/Spark2x/spark/bin/beeline -u "jdbc:hive2://<zkNode1_IP>:<zkNode1_Port>,<zkNode2_IP>:<zkNode2_Port>,<zkNode3_IP>:<zkNode3_Port>/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver2x;user.principal=spark2x/hadoop.*<system domain name>*@*<system domain name>*;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.*<system domain name>*@*<system domain name>*;"

n. Run the following command to use a SparkSQL statement to query HBase table data:

select * from hbaseTable;

12.23.2.6.5 Configuring the Client and Server

This section describes how to configure SparkSQL permission management functions (client configuration is similar to server configuration). To enable table permission, add following configurations on the client and server:

- **spark-defaults.conf** configuration file

  Table 12-383 Parameter description (1)

Parameter	Description	Default Value
spark.sql.authorization.enabled	Specifies whether to enable permission authentication of the datasource statement. It is recommended that the parameter value be set to **true** to enable permission authentication.	true

- **hive-site.xml** configuration file

Table 12-384 Parameter description (2)

Parameter	Description	Default Value
hive.metastore.uris	Specifies the MetaStore service address of the Hive component, for example, **thrift://10.10.169.84:21088,thrift://10.10.81.37:21088**.	-
hive.metastore.sasl.enabled	Specifies whether the MetaStore service uses SASL to improve security. The table permission function must be enabled.	true
hive.metastore.kerberos.principal	Specifies the principal of the MetaStore service in the Hive component, for example, **hive/hadoop.**<*system domain name*>@<*system domain name*>.	hive-metastore/_HOST@EXAMPLE.COM
hive.metastore.thrift.sasl.qop	After the SparkSQL permission management function is enabled, set the parameter to **auth-conf**.	auth-conf
hive.metastore.token.signature	Specifies the token identifier of the MetaStore service, which is set to **HiveServer2ImpersonationToken**.	HiveServer2ImpersonationToken
hive.security.authenticator.manager	Specifies the manager authenticated by the Hive client, which is set to **org.apache.hadoop.hive.ql.security.SessionStateUserGroupAuthenticator**.	org.apache.hadoop.hive.ql.security.SessionStateUserMSGroupAuthenticator
hive.security.authorization.enabled	Specifies whether to enable client authentication, which is set to **true**.	true
hive.security.authorization.createtable.owner.grants	Specifies which permissions are granted to the owner who creates the table, which is set to **ALL**.	ALL

- **core-site.xml** configuration file of the MetaStore service

Table 12-385 Parameter description (3)

Parameter	Description	Default Value
hadoop.proxyuser.spark.hosts	Specifies the hosts from which Spark users can be masqueraded, which is set to *****, indicating all hosts.	-
hadoop.proxyuser.spark.groups	Specifies the user groups from which Spark users can be masqueraded, which is set to *****, indicating all user groups.	-

12.23.2.7 Scenario-Specific Configuration

12.23.2.7.1 Configuring Multi-active Instance Mode

Scenarios

In this mode, multiple ThriftServers coexist in the cluster and the client can randomly connect any ThriftServer to perform service operations. When one or multiple ThriftServers stop working, a client can connect to another functional ThriftServer.

Configuration Description

Log in to Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations**, click **All Configurations**, and search for and modify the following parameters.

Table 12-386 Parameter description

Parameter	Description	Default Value
spark.thriftserver.zookeeper.connection.timeout	Specifies the timeout interval of connection between ZooKeeper client and ThriftServer. The unit is millisecond.	60000
spark.thriftserver.zookeeper.session.timeout	Specifies the timeout interval of a ZooKeeper client session. The unit is millisecond.	90000
spark.thriftserver.zookeeper.retry.times	Specifies the retry times after ZooKeeper disconnection.	3
spark.yarn.queue	Specifies the Yarn queue where the JDBCServer service resides.	default

12.23.2.7.2 Configuring the Multi-tenant Mode

Scenarios

In multi-tenant mode, JDBCServers are bound with tenants. Each tenant corresponds to one or more JDBCServers, and a JDBCServer provides services for only one tenant. Different tenants can be configured with different Yarn queues to implement resource isolation.

Configuration Description

Log in to Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations**, click **All Configurations**, and search for and modify the following parameters.

Table 12-387 Parameter description

Parameter	Description	Default Value
spark.proxyserver.hash.enabled	Specifies whether to connect to ProxyServer using the Hash algorithm. ● **true** indicates using the Hash algorithm. In multi-tenant mode, this parameter must be configured to **true**. ● **false** indicates using random connection. In multi-active instance mode, this parameter must be configured to **false**.	true **NOTE** After this parameter is modified, you need to download the client again.
spark.thriftserver.proxy.enabled	Specifies whether to use the multi-tenant mode. ● **false**: The multi-instance mode is used. ● **true**: The multi-tenant mode is used.	true
spark.thriftserver.proxy.maxThriftServerPerTenancy	Specifies the maximum number of JDBCServer instances that can be started by a tenant in multi-tenant mode.	1
spark.thriftserver.proxy.maxSessionPerThriftServer	Specifies the maximum number of sessions in a single JDBCServer instance in multi-tenant mode. If the number of sessions exceeds this value and the number of JDBCServer instances does not exceed the upper limit, a new JDBCServer instance is started. Otherwise, an alarm log is output.	50

Parameter	Description	Default Value
spark.thriftserver.proxy.sessionWaitTime	Specifies the wait time before a JDBCServer instance is stopped when it has no session connections in multi-tenant mode.	180000
spark.thriftserver.proxy.sessionThreshold	In multi-tenant mode, when the session usage (formula: number of current sessions/ spark.thriftserver.proxy.maxSessionPerThriftServer x number of current JDBCServer instances) of the JDBCServer instance reaches the threshold, a new JDBCServer instance is automatically added.	100
spark.thriftserver.proxy.healthcheck.period	Specifies the period of JDBCServer health checks conducted by the JDBCServer proxy in multi-tenant mode.	60000
spark.thriftserver.proxy.healthcheck.recheckTimes	Specifies the number of JDBCServer health check retries conducted by the JDBCServer proxy in multi-tenant mode.	3
spark.thriftserver.proxy.healthcheck.waitTime	Specifies the wait time for JDBCServer to respond to a health check request sent by the JDBCServer proxy.	10000
spark.thriftserver.proxy.session.check.interval	Specifies the period of JDBCServer proxy sessions in multi-tenant mode.	6h
spark.thriftserver.proxy.idle.session.timeout	Specifies the idle time interval of a JDBCServer proxy session in multi-tenant mode. If no operation is performed within this period, the session is closed.	7d
spark.thriftserver.proxy.idle.session.check.operation	Specifies whether to check that operations still exist on a JDBCServer proxy session when the session is checked for expiration in multi-tenant mode.	true
spark.thriftserver.proxy.idle.operation.timeout	Specifies the timeout interval of an operation in multi-tenant mode. An operation that times out is closed.	5d

12.23.2.7.3 Configuring the Switchover Between the Multi-active Instance Mode and the Multi-tenant Mode

Scenarios

When using a cluster, if you want to switch between multi-active instance mode and multi-tenant mode, the following configurations are required.

- Switch from multi-tenant mode to multi-active instance mode.

  Modify the following parameters of the Spark2x service:

  - spark.thriftserver.proxy.enabled=false
  - spark.scheduler.allocation.file=#{conf_dir}/fairscheduler.xml
  - spark.proxyserver.hash.enabled=false
- Switch from multi-active instance mode to multi-tenant mode.

  Modify the following parameters of the Spark2x service:

  - spark.thriftserver.proxy.enabled=true
  - spark.scheduler.allocation.file=./__spark_conf__/__hadoop_conf__/fairscheduler.xml
  - spark.proxyserver.hash.enabled=true

Configuration Description

Log in to Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Spark2x** > **Configuration**, click **All Configurations**, and search for and modify the following parameters.

Table 12-388 Parameter description

Parameter	Description	Default Value
spark.thriftserver.proxy.enabled	Specifies whether to use the multi-tenant mode. • **false**: The multi-instance mode is used. • **true**: The multi-tenant mode is used.	true
spark.scheduler.allocation.file	Specifies the fair scheduling file path. • If the multi-active instance mode is used, the path is changed to **#{conf_dir}/fairscheduler.xml**. • If multi-tenant mode is used, the path is changed to **./__spark_conf__/__hadoop_conf__/fairscheduler.xml**.	./__spark_conf__/__hadoop_conf__/fairscheduler.xml

Parameter	Description	Default Value
spark.proxyserver.hash.enabled	Specifies whether to connect to ProxyServer using the Hash algorithm. • **true** indicates using the Hash algorithm. In multi-tenant mode, this parameter must be configured to **true**. • **false** indicates using random connection. In multi-active instance mode, this parameter must be configured to **false**.	true **NOTE** After this parameter is modified, you need to download the client again.

12.23.2.7.4 Configuring the Size of the Event Queue

Scenarios

Functions such as UI, EventLog, and dynamic resource scheduling in Spark are implemented through event transfer. Events include SparkListenerJobStart and SparkListenerJobEnd, which record each important process.

Each event is saved to a queue after it occurs. When creating a SparkContext object, Driver starts a thread to obtain an event from the queue in sequence and sends the event to each Listener. Each Listener processes the event after detecting the event.

Therefore, when the queuing speed is faster than the read speed, the queue overflows. As a result, the overflow event is lost, affecting the UI, EventLog, and dynamic resource scheduling functions. Therefore, a configuration item is added for more flexible use. You can set a proper value based on the memory size of the driver.

Configuration Description

Navigation path for setting parameters:

Before executing an application, modify the Spark service configuration. On Manager, choose **Cluster** > ***Name of the desired cluster*** > **Service** > **Spark2x** > **Configuration** and click **All Configurations**. Enter a parameter name in the search box.

Table 12-389 Parameter description

Parameter	Description	Default Value
spark.scheduler.listenerbus.eventqueue.capacity	Specifies the size of the event queue. Configure this parameter based on the memory of the driver.	1000000

NOTE

If the following information is displayed in the Driver log, the queue overflows.

1. Common application:
   ```
   Dropping SparkListenerEvent because no remaining room in event queue.
   This likely means one of the SparkListeners is too slow and cannot keep
   up with the rate at which tasks are being started by the scheduler.
   ```
2. Spark Streaming application:
   ```
   Dropping StreamingListenerEvent because no remaining room in event queue.
   This likely means one of the StreamingListeners is too slow and cannot keep
   up with the rate at which events are being started by the scheduler.
   ```

12.23.2.7.5 Configuring Executor Off-Heap Memory

Scenario

When the executor off-heap memory is too small, or processes with higher priority preempt resources, the physical memory usage will exceed the maximal value. To prevent the physical memory usage from exceeding, set the following parameter.

Configuration

Navigation path for setting parameters:

When submitting an application, set the following parameter using **--conf** or adjust the parameter in the **spark-defaults.conf** configuration file on the client.

Table 12-390 Parameter description

Parameter	Description	Default Value
spark.executor.memoryOverhead	Indicates the off-heap memory of each executor, in MB. Increasing the value of this parameter prevents the physical memory usage from exceeding the maximal value. The value is calculated based on max(384, Executor – Memory x 0.1). The minimal value is 384.	1024

12.23.2.7.6 Enhancing Stability in a Limited Memory Condition

Scenario

A large amount of memory is required when Spark SQL executes a query, especially during Aggregate and Join operations. If the memory is limited, OutOfMemoryError may occur. Stability in a limited memory condition ensures queries to be run in limited memory without OutOfMemoryError.

NOTE

Limited memory does not mean infinitely small memory, but ensures stable queries by using disks in a scenario where memory fails to store the data amount that is several times larger than the available memory size. For example, for queries involving Join, the data of the same key used for Join needs to be stored in memory. If the data amount is too large to be stored in the available memory, OutOfMemoryError occurs.

Stability in a limited memory condition involves the following sub-functions:

1. ExternalSort

   If the memory is inadequate during sorting, partial data overflows to disks.

2. TungstenAggregate

   By default, ExternalSort is used to sort data before data aggregation. Therefore, if the memory is inadequate, the data overflows to disks during sorting. The data has been properly sorted before aggregation and only aggregation results of the current key are remained, which use a small amount of memory.

3. SortMergeJoin and SortMergeOuterJoin

   SortMergeJoin and SortMergeOuterJoinan are based on the equivalence join of sorted data. By default, ExternalSort is used to sort the data before the equivalence join. Therefore, if the memory is inadequate, the data overflows to disks during sorting. The data has been properly sorted before the equivalence join and only the data of the same key are remained, which uses a small amount of memory.

Configuration

Navigation path for setting parameters:

When submitting an application, set the following parameters using **--conf** or adjust the parameters in the **spark-defaults.conf** configuration file on the client.

Table 12-391 Parameter description

Parameter	Scenario	Description	Default Value
spark.sql.tungsten.enabled	/	Type: Boolean ● If the value is **true**, tungsten is enabled. That is, the logic plan is equivalent to the codegeneration function, and the physical plan uses the corresponding tungsten execution plan. ● If the value is **false**, tungsten is disabled.	true

Parameter	Scenario	Description	Default Value
spark.sql.codegen.wholeStage		Type: Boolean • If the value is **true**, codegeneration is enabled. That is, for some specified queries, the logic plan code will be generated dynamically when running. • If the value is **false**, codegeneration is disabled and the existing static code is used.	true

NOTE

1. To enable ExternalSort, you need to set **spark.sql.planner.externalSort** to **true** and **spark.sql.unsafe.enabled** to **false** or **spark.sql.codegen.wholeStage** to **false**.
2. To enable TungstenAggregate, use either of the following methods:

   Set **spark.sql.codegen.wholeStage** and **spark.sql.unsafe.enabled** to **true** in the configuration file or CLI.

   If neither **spark.sql.codegen.wholeStage** nor **spark.sql.unsafe.enabled** is **true** or either of them is **true**, TungstenAggregate is enabled as long as **spark.sql.tungsten.enabled** is set to **true**.

12.23.2.7.7 Viewing Aggregated Container Logs on the Web UI

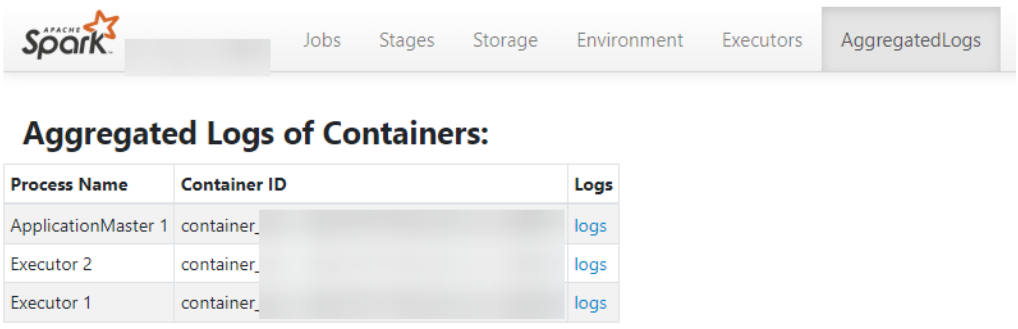
Scenarios

When **yarn.log-aggregation-enable** of Yarn is set to **true**, the container log aggregation function is enabled. Log aggregation indicates that after applications are run on Yarn, NodeManager aggregates all container logs of the node to HDFS and deletes local logs. For details, see **Configuring Container Log Aggregation**.

However, all logs will be aggregated to an HDFS directory and can only be viewed by accessing an HDFS file. Open-source Spark and Yarn do not support the function of viewing aggregated logs on the web UI.

Spark supports this function. As shown in **Figure 12-54**, the **AggregatedLogs** tab is added to the HistoryServer page. You can click **logs** to view aggregated logs.

Figure 12-54 Log aggregation page



Configuration Description

To display logs on the web UI, aggregated logs need to be parsed and presented. Spark parses aggregation logs using JobHistoryServer of Hadoop. Therefore, you can use the **spark.jobhistory.address** parameter to specify the URL of the JobHistoryServer page to parse and present the logs.

Navigation path for setting parameters:

When submitting an application, set these parameters using **--conf** or adjust the following parameter in the **spark-defaults.conf** configuration file on the client.

NOTE

- This function depends on JobHistoryServer of Hadoop. Therefore, ensure that JobHistoryServer is running properly before using the log aggregation function.
- If the parameter value is empty, the **AggregatedLogs** tab page still exists, but you cannot view logs by clicking **logs**.
- The aggregated container logs can be viewed only when the application is running and event log files of the application exist on HDFS.
- You can click the log link on the **Executors** page to view the logs of a running task. After the task completes, the logs are aggregated to HDFS, and the log link on the **Executors** page becomes invalid. In this case, you can click **logs** on the **AggregatedLogs** page to view the aggregated logs.

Table 12-392 Parameter description

Parameter	Description	Default Value
spark.jobhistory.address	URL of the JobHistoryServer page. The format is *http(s)://ip:port/jobhistory*. For example, **https://10.92.115.1:26014/jobhistory**. The default value is empty, indicating that container aggregation logs cannot be viewed on the web UI. Restart the service for the configuration to take effect.	-

12.23.2.7.8 Configuring Environment Variables in Yarn-Client and Yarn-Cluster Modes

Scenario

Values of some configuration parameters of Spark client vary depending on its work mode (YARN-Client or YARN-Cluster). If you switch Spark client between different modes without first changing values of such configuration parameters, Spark client fails to submit jobs in the new mode.

To avoid this, configure parameters as described in **Table 12-393**.

- In Yarn-Cluster mode, use the new parameters (path and parameters of Spark server).
- In Yarn-Client mode, uses the original parameters.

  They are **spark.driver.extraClassPath**, **spark.driver.extraJavaOptions**, and **spark.driver.extraLibraryPath**.

NOTE

If you choose not to add the parameters in **Table 12-393**, Spark client can continue to operate well in either mode but the mode switch requires changes to some of its configuration parameters.

Configuration Parameters

Navigation path for setting parameters:

On Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations**. Click **All Configurations** and enter a parameter name in the search box.

Table 12-393 Parameter description

Parameter	Description	Default Value
spark.yarn.cluster.driver.extraClassPath	Indicates the extraClassPath of the driver in Yarn-cluster mode. Set the parameter to the path and parameters of the server. The original parameter **spark.driver.extraClassPath** indicates the extraClassPath of Spark client. By using different parameters to separate the settings of Spark server from the settings of Spark client, you can switch Spark client to different modes without changing parameter values.	${BIGDATA_HOME}/common/runtime/security

Parameter	Description	Default Value
spark.yarn.cluster.driver.extraJavaOptions	Indicates the extraJavaOptions of Driver in Yarn-Cluster mode and is set to path and parameters of extraJavaOptions of Spark server. The original parameter **spark.driver.extraJavaOptions** indicates the path of extraJavaOptions of Spark client. By using different parameters to separate the settings of Spark server from the settings of Spark client, you can switch Spark client to different modes without changing parameter values.	-Xloggc:<LOG_DIR>/indexserver-%p-gc.log -XX:+PrintGCDetails -XX:-OmitStackTraceInFastThrow -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=20 -XX:GCLogFileSize=10M -Dlog4j.configuration=./__spark_conf__/__hadoop_conf__/log4j-executor.properties -Dlog4j.configuration.watch=true -Djava.security.auth.login.config=./__spark_conf__/__hadoop_conf__/jaas-zk.conf -Dzookeeper.server.principal=${ZOOKEEPER_SERVER_PRINCIPAL} -Djava.security.krb5.conf=./__spark_conf__/__hadoop_conf__/kdc.conf -Djetty.version=x.y.z -Dorg.xerial.snappy.tempdir=${BIGDATA_HOME}/tmp -Dcarbon.properties.filepath=./__spark_conf__/__hadoop_conf__/carbon.properties -Djdk.tls.ephemeralDHKeySize=2048 -Dspark.ssl.keyStore=./child.keystore #{java_stack_prefer}

12.23.2.7.9 Configuring the Default Number of Data Blocks Divided by SparkSQL

Scenarios

By default, SparkSQL divides data into 200 data blocks during shuffle. In data-intensive scenarios, each data block may have excessive size. If a single data block of a task is larger than 2 GB, an error similar to the following will be reported while Spark attempts to fetch the data block:

```
Adjusted frame length exceeds 2147483647: 2717729270 - discarded
```

For example, setting the number of default data blocks to 200 causes SparkSQL to encounter an error in running a TPCDS 500-GB test. To avoid this, increase the number of default blocks in data-intensive scenarios.

Configuration parameters

Navigation path for setting parameters:

On Manager, choose **Cluster** > ***Name of the desired cluster*** > **Service** > **Spark2x** > **Configuration** and click **All Configurations**. Enter a parameter name in the search box.

Table 12-394 Parameter description

Parameter	Description	Default Value
spark.sql.shuffle.partitions	Indicates the default number of blocks divided during shuffle.	200

12.23.2.7.10 Configuring the Compression Format of a Parquet Table

Scenarios

The compression format of a Parquet table can be configured as follows:

1. If the Parquet table is a partitioned one, set the **parquet.compression** parameter of the Parquet table to specify the compression format. For example, set **tblproperties** in the table creation statement: **"parquet.compression"="snappy"**.
2. If the Parquet table is a non-partitioned one, set the **spark.sql.parquet.compression.codec** parameter to specify the compression format. The configuration of the **parquet.compression** parameter is invalid, because the value of the **spark.sql.parquet.compression.codec** parameter is read by the **parquet.compression** parameter. If the **spark.sql.parquet.compression.codec** parameter is not configured, the default value is **snappy** and will be read by the **parquet.compression** parameter.

Therefore, the **spark.sql.parquet.compression.codec** parameter can only be used to set the compression format of a non-partitioned Parquet table.

Configuration parameters

Navigation path for setting parameters:

On Manager, choose **Cluster** > ***Name of the desired cluster*** > **Service** > **Spark2x** > **Configuration**. Click **All Configurations** and enter a parameter name in the search box.

Table 12-395 Parameter description

Parameter	Description	Default Value
spark.sql.parquet.compression.codec	Used to set the compression format of a non-partitioned Parquet table.	snappy

12.23.2.7.11 Configuring the Number of Lost Executors Displayed in WebUI

Scenario

In Spark WebUI, the **Executor** page can display information about Lost Executor. Executors are dynamically recycled. If the JDBCServer tasks are large, there may be too many lost executors displayed in WebUI. Therefore, the number of displayed lost executors can be configured.

Procedure

Configure the following parameter in the **spark-defaults.conf** file on Spark client.

Table 12-396 Parameter description

Parameter	Description	Default Value
spark.ui.retainedDeadExecutors	The maximum number of Lost Executors displayed in Spark WebUI.	100

12.23.2.7.12 Setting the Log Level Dynamically

Scenarios

In some scenarios, to locate problems or check information by changing the log level,

you can add the **-Dlog4j.configuration.watch=true** parameter to the JVM parameter of a process before the process is started. After the process is started, you can modify the log4j configuration file corresponding to the process to change the log level.

The following processes support the dynamic setting of log levels: driver, executor, ApplicationMaster, JobHistory and JDBCServer.

Allowed log levels are as follows: FATAL, ERROR, WARN, INFO, DEBUG, TRACE, and ALL.

Configuration Description

Add the following parameters to the JVM parameter corresponding to a process.

Table 12-397 Parameter description

Parameter	Description	Default Value
-Dlog4j.configuration.watch	Indicates a JVM parameter of a process. If this parameter is set to **true**, the dynamic configuration of log levels is enabled.	Left blank, indicating that the dynamic configuration of log levels is disabled

Table 12-398 lists the JVM parameters of the driver, executor, and ApplicationMaster processes. Configure the following parameters in the **spark-defaults.conf** file on the Spark client. Set the log levels of the driver, executor, and ApplicationMaster processes in the log4j configuration file specified by the **-Dlog4j.configuration** parameter.

Table 12-398 JVM parameters of processes (1)

Parameter	Description	Default Log Level
spark.driver.extraJavaOptions	Indicates the JVM parameter of the driver process.	INFO
spark.executor.extraJavaOptions	Indicates the JVM parameter of the executor process.	INFO
spark.yarn.am.extraJavaOptions	Indicates the JVM parameter of the ApplicationMaster process.	INFO

Table 12-399 describes the JVM parameters of JobHistory Server and JDBCServer. Set the parameters in the **ENV_VARS** configuration file. Set the log levels of JobHistory Server and JDBCServer in the **log4j.properties** configuration file.

Table 12-399 JVM parameters of processes (2)

Parameter	Description	Default Log Level
GC_OPTS	Indicates the JVM parameter of the JobHistory Server process.	INFO
SPARK_SUBMIT_OPTS	Indicates the JVM parameter of JDBCServer.	INFO

Example:

To change the log level of the executor process to DEBUG dynamically, modify the **spark.executor.extraJavaOptions** JVM parameter of the executor process in the **spark-defaults.conf** file and run the following command to add the following configuration before the process is started:

```
-Dlog4j.configuration.watch=true
```

After the user application is submitted, change the log level in the log4j configuration file (for example, **-Dlog4j.configuration=file:${BIGDATA_HOME}/FusionInsight_Spark2x_8.1.0.1/install/FusionInsight-Spark2x-3.1.1/spark/conf/log4j-executor.properties**) specified by the **-Dlog4j.configuration** parameter in **spark.executor.extraJavaOptions** to DEBUG:

```
log4j.rootCategory=DEBUG, sparklog
```

It takes several seconds for the DEBUG level to take effect.

12.23.2.7.13 Configuring Whether Spark Obtains HBase Tokens

Scenario

When Spark is used to submit tasks, the driver obtains tokens from HBase by default. To access HBase, you need to configure the **jaas.conf** file for security authentication. If the **jaas.conf** file is not configured, the application will fail to run.

Therefore, perform the following operations based on whether the application involves HBase:

- If the application does not involve HBase, you do not need to obtain the HBase tokens. In this case, set **spark.yarn.security.credentials.hbase.enabled** to **false**.
- If the application involves HBase, set **spark.yarn.security.credentials.hbase.enabled** to **true** and configure the **jaas.conf** file on the driver as follows:

  ```
  {client}/spark/bin/spark-sql  --master yarn-client --principal {principal} --keytab {keytab} --driver-java-options "-Djava.security.auth.login.config={LocalPath}/jaas.conf"
  ```

  Specify Keytab and Principal in the **jaas.conf** file. The following is an example:

  ```
  Client {
  com.sun.security.auth.module.Krb5LoginModule required
  useKeyTab=true
  keyTab = "{LocalPath}/user.keytab"
  principal="super@<System domain name>"
  useTicketCache=false
  debug=false;
  };
  ```

Configuration

Configure the following parameter in the **spark-defaults.conf** file of the Spark client.

Table 12-400 Parameter description

Parameter	Description	Default Value
spark.yarn.security.credentials.hbase.enabled	Indicates whether HBase obtains a token. • **true**: HBase obtains a token. • **false**: HBase does not obtain a token.	false

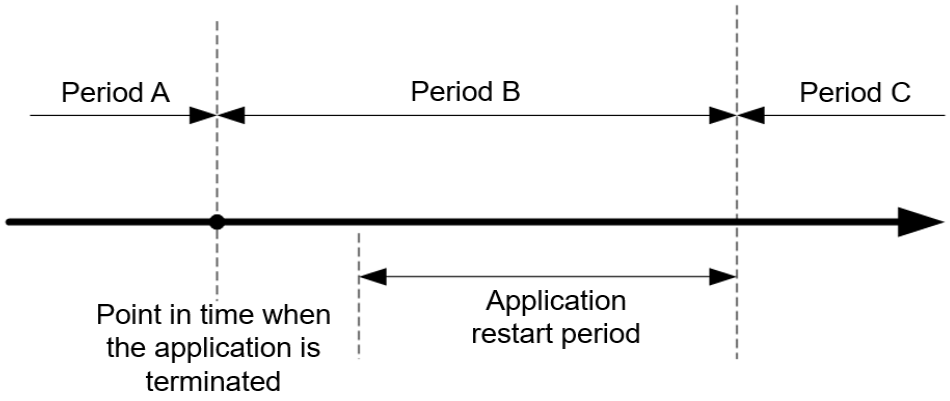
12.23.2.7.14 Configuring LIFO for Kafka

Scenario

If the Spark Streaming application is connected to Kafka, after the Spark Streaming application is terminated abnormally and restarted from the checkpoint, the system preferentially processes the tasks that are not completed before the application is terminated (Period A) and the tasks generated based on data that enters Kafka during the period (Period B) from the application termination to the restart. Then the application processes the tasks generated based on data that enters Kafka after the application is restarted (Period C). For data that enters Kafka in period B, Spark generates a corresponding number of tasks based on the end time (**batch** time). The first task reads all data, but other tasks may not read data. As a result, the task processing pressure is uneven.

If the tasks in Period A and Period B are processed slowly, the processing of tasks in period C is affected. To cope with the preceding scenario, Spark provides the last-in first-out (LIFO) function for Kafka.

Figure 12-55 Time axis for restarting the Spark Streaming application



After this function is enabled, Spark preferentially schedules tasks in Period C. If there are multiple tasks in Period C, Spark schedules and executes the tasks in the sequence of task generation. Then Spark executes the tasks in Periods A and B. For data that enters Kafka in Period B, Spark generates tasks based on the end time

and evenly distributes all data that enters Kafka in this period to each task to avoid uneven task processing pressure.

Constraints:

- This function applies only to the direct mode of Spark Streaming, and the execution result does not depend on the processing result of the previous batch (that is, stateless operation, for example, **updatestatebykey**). Multiple data input streams must be comparatively independent from each other. Otherwise, the result may change after the data is divided.
- The Kafka LIFO function can be enabled only when the application is connected to the Kafka input source.
- If both Kafka LIFO and flow control functions are enabled when the application is submitted, the flow control function is not enabled for the data that enters Kafka in Period B to ensure that the task scheduling priority for reading the data is the lowest. Flow control is enabled for the tasks in Period C after the application is restarted.

Configuration

Configure the following parameters in the **spark-defaults.conf** file on the Spark driver.

Table 12-401 Parameter description

Parameter	Description	Default Value
spark.streaming.kafka.direct.lifo	Specifies whether to enable the LIFO function of Kafka.	false
spark.streaming.kafka010.inputstream.class	Obtains the decoupled class on FusionInsight.	org.apache.spark.streaming.kafka010.HWDirectKafkaInputDStream

12.23.2.7.15 Configuring Reliability for Connected Kafka

Scenario

When the Spark Streaming application is connected to Kafka and the application is restarted, the application reads data from Kafka based on the last read topic offset and the latest offset of the current topic.

If the leader of a Kafka topic fails and the offset of the Kafka leader is greatly different from that of the Kafka follower, the Kafka follower and leader are switched over after the Kafka service is restarted. As a result, the offset of the topic decreases after the Kafka service is restarted.

- If the Spark Streaming application keeps running, the start position for reading Kafka data is greater than the end position because the offset of the topic in Kafka decreases. As a result, the application cannot read data from Kafka and reports an error.

- Before restarting the Kafka service, stop the Spark Streaming application. After the Kafka service is restarted, restart the Spark Streaming application to restore the application from the checkpoint. In this case, the Spark Streaming application records the offset position read before the termination and uses the position as the reference to read subsequent data. The Kafka offset decreases (for example, from 100,000 to 10,000). Spark Streaming consumes data only after the offset of the Kafka leader increases to 100,000. As a result, the newly sent data whose offset is between 10,000 and 100,000 is lost.

To resolve the preceding problem, you can configure reliability for Kafka connected to Spark Streaming. After the reliability function of connected Kafka is enabled:

- If the offset of a topic in Kafka decreases when the Spark Streaming application is running, the latest offset of the topic in Kafka is used as the start position for reading Kafka data and subsequent data is read.

  For a task that has been generated but has not been scheduled, if the read Kafka offset is greater than the latest offset of the topic in Kafka, the task fails to be executed.

  NOTE

  If a large number of tasks fail, the Executor is added to the blacklist. As a result, subsequent tasks cannot be deployed and run. If this happens, you can set **spark.blacklist.enabled** to disable the blacklist function. The blacklist function is enabled by default.

- If the offset of a topic in Kafka decreases, the Spark Streaming application restarts to restore the unfinished tasks. If the read Kafka offset range is greater than the latest offset of the topic in Kafka, the task is directly discarded.

NOTE

If the state function is used in the Spark Streaming application, do not enable the reliability function of connected Kafka.

Configuration

Configure the following parameter in the **spark-defaults.conf** file of the Spark client.

Table 12-402 Parameter description

Parameter	Description	Default Value
spark.streaming.Kafka.reliability	Indicates whether to enable the reliability function for Kafka connected to Spark Streaming. • **true**: The reliability function is enabled. • **false**: The reliability function is disabled.	false

12.23.2.7.16 Configuring Streaming Reading of Driver Execution Results

Scenario

When a query statement is executed, the returned result may be large (containing more than 100,000 records). In this case, JDBCServer out of memory (OOM) may occur. Therefore, the data aggregation function is provided to avoid OOM without sacrificing the performance.

Configuration

Two data aggregation function configuration parameters are provided. The two parameters are set in the **tunning** option on the Spark JDBCServer server. After the setting is complete, restart JDBCServer.

Table 12-403 Parameter description

Parameter	Description	Default Value
spark.sql.bigdata.thriftServer.useHdfsCollect	Indicates whether to save result data to HDFS instead of the memory. Advantages: The query result is stored in HDFS. Therefore, JDBCServer OOM does not occur. Disadvantages: The query is slow. ● **true**: Result data is saved to HDFS. ● **false**: This function is disabled. **NOTICE** When **spark.sql.bigdata.thriftServer.useHdfsCollect** is set to **true**, result data is saved to HDFS. However, the job description on the native JobHistory page cannot be associated with the corresponding SQL statement. In addition, the execution ID in the spark-beeline command output is null. To solve the JDBCServer OOM problem and ensure correct information display, you are advised to set **spark.sql.userlocalFileCollect**.	false

Parameter	Description	Default Value
spark.sql.uselocalFileCollect	Indicates whether to save result data to the local disk instead of memory. Advantages: In the case of small data volume, the performance loss can be ignored compared with the data storage mode using the native memory. In the case of large data volume (hundreds of millions of data records), the performance is much better than that when data is stored in the HDFS and native memory. Disadvantages: Optimization is required. In the case of large data volume, it is recommended that the JDBCServer driver memory be 10 GB and each core of the executor be allocated with 3 GB memory. ● **true**: This function is enabled. ● **false**: This function is disabled.	false
spark.sql.collect.Hive	This parameter is valid only when **spark.sql.uselocalFileCollect** is set to **true**. It indicates whether to save the result data to a disk in direct serialization mode or in indirect serialization mode. Advantage: For queries of tables with a large number of partitions, the aggregation performance of the query results is better than that of the storage mode that query results are directly stored on the disk. Disadvantages: The disadvantages are the same as those when **spark.sql.uselocalFileCollect** is enabled. ● **true**: This function is enabled. ● **false**: This function is disabled.	false
spark.sql.collect.serialize	This parameter takes effect only when both **spark.sql.uselocalFileCollect** and **spark.sql.collect.Hive** are set to **true**. The function is to further improve performance. ● **java**: Data is collected in Java serialization mode. ● **kryo**: Data is collected in kryo serialization mode. The performance is better than that when the Java serialization mode is used.	java

NOTE

spark.sql.bigdata.thriftServer.useHdfsCollect and **spark.sql.uselocalFileCollect** cannot be set to **true** at the same time.

12.23.2.7.17 Filtering Partitions without Paths in Partitioned Tables

Scenario

When you perform the ***select*** query in Hive partitioned tables, the **FileNotFoundException** exception is displayed if a specified partition path does not exist in HDFS. To avoid the preceding exception, configure **spark.sql.hive.verifyPartitionPath** parameter to filter partitions without paths.

Procedure

Perform either of the following methods to filter partitions without paths:

- Configure the following parameter in the **spark-defaults.conf** file on Spark client.

  Table 12-404 Parameter description

Parameter	Description	Default Value
spark.sql.hive.verifyPartitionPath	Whether to filter partitions without paths when reading Hive partitioned tables. **true**: enables the filtering **false**: disables the filtering	false

- When running the spark-submit command to submit an application, configure the **--conf** parameter to filter partitions without paths.

  For example:

  ```
  spark-submit --class org.apache.spark.examples.SparkPi  --conf spark.sql.hive.verifyPartitionPath=true $SPARK_HOME/lib/spark-examples_*.jar
  ```

12.23.2.7.18 Configuring Spark2x Web UI ACLs

Scenario

Users need to implement security protection for Spark2x web UI when some data on the UI cannot be viewed by other users. Once a user attempts to log in to the UI, Spark2x can check the view ACL of the user to determine whether to allow the user to access the UI.

Spark2x has two types of web UI. One is for running tasks. You can access the web UI using the application link on the native Yarn page or the REST APIs. The other one is for ended tasks. You can access the web UI using the Spark2x JobHistory service or the REST APIs.

NOTE

This section applies only to clusters in security mode (with Kerberos authentication enabled).

- Configuring the ACL of the web UI for running tasks

  For a running task, you can set the following parameters on the server:

  - **spark.admin.acls**: specifies the web UI administrator list.
  - **spark.admin.acls.groups**: specifies the administrator group list.
  - **spark.ui.view.acls**: specifies the Yarn page visitor list.
  - **spark.modify.acls.groups**: specifies the Yarn page visitor group list.
  - **spark.modify.acls**: specifies the web UI modifier list.
  - **spark.ui.view.acls.groups**: specifies the web UI modifier group list.

- Configuring the ACL of the web UI for ended tasks

  For ended tasks, use client parameter **spark.history.ui.acls.enable** to enable or disable the ACL access permission.

  If ACL control is enabled, configure client parameters **spark.admin.acls** and **spark.admin.acls.groups** to specify the web UI administrator list and administrator group list. Use client parameters **spark.ui.view.acls** and **spark.modify.acls.groups** to specify the visitor list and visitor group list that view web UI task details. Use client parameters **spark.modify.acls** and **spark.ui.view.acls.groups** to specify the visitor list and group list that modify web UI task details.

Configuration

Log in to FusionInsight Manager, choose **Cluster** > ***Name of the desired cluster*** > **Services** > **Spark2x** > **Configurations**, click **All Configurations**, search for **acl**, and modify the following parameters on the JobHistory, JDBCServer, SparkResource, and Spark pages.

Table 12-405 Parameter description

Parameter	Description	Default Value
spark.history.ui.acls.enable	Indicates whether JobHistory supports the permission verification of a single task.	true
spark.acls.enable	Indicates whether to enable Spark permission management. If this function is enabled, the system checks whether the user has the permission to access and modify task information.	true

Parameter	Description	Default Value
spark.admin.acls	Indicates the list of Spark administrators. All members in the list have the rights to manage all Spark tasks. You can configure multiple administrators and separate them from each other using commas (,).	admin
spark.admin.acls.groups	Indicates the list of Spark administrator groups. All groups in the list have the permission to manage all Spark tasks. You can configure multiple administrator groups and separate them from each other using commas (,).	-
spark.modify.acls	Indicates the list of members that have the permission to modify Spark tasks. By default, the user who starts a task has the permission to modify the task. You can configure multiple users and separate them from each other using commas (,).	-
spark.modify.acls.groups	Indicates the list of groups that have the permission to modify Spark tasks. You can configure multiple groups and separate them from each other using commas (,).	-
spark.ui.view.acls	Indicates the list of members that have the permission to access Spark tasks. By default, the user who starts a task has the permission to modify the task. You can configure multiple users and separate them from each other using commas (,).	-
spark.ui.view.acls.groups	Indicates the list of groups that have the permission to access Spark tasks. You can configure multiple groups and separate them from each other using commas (,).	-

NOTE

If you use a client to submit tasks, you must download the client again after modifying the **spark.admin.acls**, **spark.admin.acls.groups**, **spark.modify.acls**, **spark.modify.acls.groups**, **spark.ui.view.acls**, and **spark.ui.view.acls.groups** parameters.

12.23.2.7.19 Configuring Vector-based ORC Data Reading

Scenario

ORC is a column-based storage format in the Hadoop ecosystem. It originates from Apache Hive and is used to reduce the Hadoop data storage space and accelerate the Hive query speed. Similar to Parquet, ORC is not a pure column-based storage format. In the ORC format, the entire table is split based on the row group, data in each row group is stored by column, and data is compressed as much as possible to reduce storage space consumption. Vector-based ORC data reading significantly improves the ORC data reading performance. In Spark2.3, SparkSQL supports vector-based ORC data reading (this function is supported in earlier Hive versions). Vector-based ORC data reading improves the data reading performance by multiple times.

This feature can be enabled by using the following parameter.

- **spark.sql.orc.enableVectorizedReader**: specifies whether vector-based ORC data reading is supported. The default value is **true**.
- **spark.sql.codegen.wholeStage**: specifies whether to compile all stages of multiple operations into a Java method. The default value is **true**.
- **spark.sql.codegen.maxFields**: specifies the maximum number of fields (including nested fields) supported by all stages of codegen. The default value is **100**.
- **spark.sql.orc.impl**: specifies whether Hive or Spark SQL native is used as the SQL execution engine to read ORC data. The default value is **hive**.

Parameters

Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x**, click the **Configurations** tab and then **All Configurations**, and search for the following parameters.

Parameter	Description	Default Value	Value Range
spark.sql.orc.enableVectorizedReader	Specifies whether vector-based ORC data reading is supported. The default value is **true**.	true	[true,false]
spark.sql.codegen.wholeStage	Specifies whether to compile all stages of multiple operations into a Java method. The default value is **true**.	true	[true,false]
spark.sql.codegen.maxFields	Specifies the maximum number of fields (including nested fields) supported by all stages of codegen. The default value is **100**.	100	Greater than 0

Parameter	Description	Default Value	Value Range
spark.sql.orc.impl	Specifies whether Hive or Spark SQL native is used as the SQL execution engine to read ORC data. The default value is **hive**.	hive	[hive,native]

NOTE

1. To use vector-based ORC data reading of SparkSQL, the following conditions must be met:
   - **spark.sql.orc.enableVectorizedReader** must be set to **true** (default value). Generally, the value is not changed.
   - **spark.sql.codegen.wholeStage** must be set to **true** (default value). Generally, the value is not changed.
   - The value of **spark.sql.codegen.maxFields** must be greater than or equal to the number of columns in scheme.
   - All data is of the AtomicType. Specifically, data is not null or of the UDT, array, or map type. If there is data of the preceding types, expected performance cannot be obtained.
   - **spark.sql.orc.impl** must be set to **native**. The default value is **hive**.
2. If a task is submitted using the client, modification of the following parameters takes effect only after you download the client again: **spark.sql.orc.enableVectorizedReader**, **spark.sql.codegen.wholeStage**, **spark.sql.codegen.maxFields**, and **spark.sql.orc.impl**.

12.23.2.7.20 Broaden Support for Hive Partition Pruning Predicate Pushdown

Scenario

In earlier versions, the predicate for pruning Hive table partitions is pushed down. Only comparison expressions between column names and integers or character strings can be pushed down. In version 2.3, pushdown of the null, in, and, or expressions are supported.

Parameters

Log in to FusionInsight Manager and choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x**. On the page that is displayed, click the **Configurations** tab then the **All Configurations** sub-tab, and search for the following parameters:

Parameter	Description	Default Value	Value Range
spark.sql.hive.advancedPartitionPredicatePushdown.enabled	Specifies whether to broaden the support for Hive partition pruning predicate pushdown.	true	[true,false]

12.23.2.7.21 Hive Dynamic Partition Overwriting Syntax

Scenario

In earlier versions, when the **insert overwrite** syntax is used to overwrite partition tables, only partitions with specified expressions are matched, and partitions without specified expressions are deleted. In Spark2.3, partitions without specified expressions are automatically matched. The syntax is the same as that of the dynamic partition matching syntax of Hive.

Parameters

Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations**, click **All Configurations**, and search for the following parameter.

Parameter	Description	Default Value	Value Range
spark.sql.sources.partitionOverwrite-Mode	Specifies the mode for inserting data in partition tables by running the **insert overwrite** command, which can be **STATIC** or **DYNAMIC**. When it is set to **STATIC**, Spark deletes all partitions based on the matching conditions. When it is set to **DYNAMIC**, Spark matches partitions based on matching conditions and dynamically matches partitions without specified conditions.	STATIC	[STATIC,DYNAMIC]

12.23.2.7.22 Configuring the Column Statistics Histogram to Enhance the CBO Accuracy

Scenarios

The execution plan for SQL statements is optimized in Spark. Common optimization rules are heuristic optimization rules. Heuristic optimization rules are provided based on the characteristics of logical plans, and the data characteristics and the execution costs of operators are not considered. Spark 2.20 introduces the Cost-Based Optimization (CBO). CBO collects statistics on tables and columns and estimates the number of output records and size of each operator in bytes based on the input data sets of operators, which is the cost of executing an operator.

CBO will adjust the execution plan to minimize the end-to-end query time. The main points are as follows:

- Filter out irrelevant data as soon as possible.

- Minimize the cost of each operator.

The CBO optimization process is divided into two steps:

1. Collect statistics.
2. Estimate the output data sets of a specific operator based on the input data sets.

Table-level statistics includes: number of records and the total size of a table data file.

Column-level statistics includes: number of unique values, maximum value, minimum value, number of null values, average length, maximum length, and the histogram.

After the statistics is obtained, the execution cost of operators can be estimated. Common operators include filter and join operators.

Histogram is a type of column statistics. It can clearly describe the distribution of column data. The column data is distributed to a specified number of bins that are displayed in ascending order by size. The upper and lower limits of each bin are calculated. The amount of data in all bins is the same (a contour histogram). After the data is distributed, the cost estimation of each operator is more accurate and the optimization effect is better.

This feature can be enabled by using the following parameter.

spark.sql.statistics.histogram.enabled: specifies whether to enable the histogram function. The default value is **false**.

Parameter Configuration

Log in to FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**, click **All Configurations**, and search for the following parameters.

Parameter	Description	Default Value	Value Range
spark.sql.cbo.enabled	Enables CBO to estimate the statistics for the execution plan.	false	[true,false]
spark.sql.cbo.joinReorder.enabled	Enables CBO join for reordering.	false	[true,false]
spark.sql.cbo.joinReorder.dp.threshold	Specifies the maximum number of nodes that can be joined in the dynamic planning algorithm.	12	>=1

Parameter	Description	Default Value	Value Range
spark.sql.cbo.joinReorder.card.weight	Specifies the proportion of dimension (number of rows) in the comparison of planned cost during reconnection: Number of rows x Proportion of dimension + File size x (1 - Proportion of dimension)	0.7	0-1
spark.sql.statistics.size.autoUpdate.enabled	Enables the function of automatically updating the table size when the table data volume changes. Note: If there are a large number of data files in a table, this operation is time consume, and the data processing speed is reduced.	false	[true,false]
spark.sql.statistics.histogram.enabled	After this function is enabled, a histogram is generated when column statistics is collected. Histogram can improve the estimation accuracy, but collecting histogram information requires additional workload.	false	[true,false]
spark.sql.statistics.histogram.numBins	Specifies the number of bins for the generated histogram.	254	>=2
spark.sql.statistics.ndv.maxError	Specifies the maximum estimation deviation allowed by the HyperLogLog++ algorithm when the column level statistics is generated.	0.05	0-1

Parameter	Description	Default Value	Value Range
spark.sql.statistics.percentile.accuracy	Specifies the accuracy rate of the percentile estimation when the contour histogram is generated. The larger the value is, the more accurate the estimation is. The estimation error value can be obtained through (1.0/ Accuracy rate of the percentile estimation).	10000	>=1

NOTE

- If you want the histogram to take effect in CBO, the following conditions must be met:
  - Set **spark.sql.statistics.histogram.enabled** to **true**. The default value is **false**. Change the value to **true** to enable the histogram function.
  - Set **spark.sql.cbo.enabled** to **true**. The default value is **false**. Change the value to **true** to enable CBO.
  - Set **spark.sql.cbo.joinReorder.enabled** to **true**. The default value is **false**. Change the value to **true** to enable connection reordering.
- If a client is used to submit a task, you need to download the client again after configuring the following parameters: **spark.sql.cbo.enabled**, **spark.sql.cbo.joinReorder.enabled**, **spark.sql.cbo.joinReorder.dp.threshold**, **spark.sql.cbo.joinReorder.card.weight**, **spark.sql.statistics.size.autoUpdate.enabled**, **spark.sql.statistics.histogram.enabled**, **spark.sql.statistics.histogram.numBins**, **spark.sql.statistics.ndv.maxError**, and **spark.sql.statistics.percentile.accuracy**.

12.23.2.7.23 Configuring Local Disk Cache for JobHistory

Scenarios

JobHistory can use local disks to cache the historical data of Spark applications to prevent the JobHistory memory from loading a large amount of application data, reducing the memory pressure. In addition, the cached data can be reused to improve the speed for subsequent application access.

Parameter Configuration

Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations**, click the **All Configurations** tab, and search for the following parameters:

Parameter	Description	Default Value
spark.history.store.path	Specifies the local directory for storing historical information for JobHistory. If this parameter is specified, JobHistory caches historical application data in the local disk instead of the memory.	${BIGDATA_HOME}/tmp/spark2x_JobHistory
spark.history.store.maxDiskUsage	Specifies the maximum available space of the local disk cache.	10 GB

12.23.2.7.24 Configuring Spark SQL to Enable the Adaptive Execution Feature

Scenario

The Spark SQL adaptive execution feature enables Spark SQL to optimize subsequent execution processes based on intermediate results to improve overall execution efficiency. The following features have been implemented:

1. Automatic configuration of the number of shuffle partitions

   Before the adaptive execution feature is enabled, Spark SQL specifies the number of partitions for a shuffle process by specifying the **spark.sql.shuffle.partitions** parameter. This method lacks flexibility when multiple SQL queries are performed on an application and cannot ensure optimal performance in all scenarios. After adaptive execution is enabled, Spark SQL automatically configures the number of partitions for each shuffle process, instead of using the general configuration. In this way, the proper number of partitions is automatically used during each shuffle process.

2. Dynamic adjusting of the join execution plan

   Before the adaptive execution feature is enabled, Spark SQL creates an execution plan based on the optimization results of rule-based optimization (RBO) and Cost-Based Optimization (CBO). This method ignores changes of result sets during data execution. For example, when a view created based on a large table is joined with other large tables, the execution plan cannot be adjusted to BroadcastJoin even if the result set of the view is small. After the adaptive execution feature is enabled, Spark SQL can dynamically adjust the execution plan based on the execution result of the previous stage to obtain better performance.

3. Automatic processing of data skew

   If data skew occurs during SQL statement execution, the memory overflow of an executor or slow task execution may occur. After the adaptive execution feature is enabled, Spark SQL can automatically process data skew scenarios. Multiple tasks are started for partitions where data skew occurs. Each task reads several output files obtained from the shuffle process and performs union operations on the join results of these tasks to eliminate data skew.

Parameters

Log in to FusionInsight Manager, choose **Cluster** > **Services** > **Spark2x** > **Configurations**, click **All Configurations**, and search for the following parameter.

Parameter	Description	Default Value
spark.sql.adaptive.enabled	Specifies whether to enable the adaptive execution function. Note: If AQE and Static Partition Pruning (DPP) are enabled at the same time, DPP takes precedence over AQE during SparkSQL task execution. As a result, AQE does not take effect.	false
spark.sql.optimizer.dynamicPartitionPruning.enabled	The switch to enable DPP.	true
spark.sql.adaptive.coalescePartitions.enabled	If this parameter is set to **true** and **spark.sql.adaptive.enabled** is set to **true**, Spark combines partitions that are consecutively random played based on the target size (specified by **spark.sql.adaptive.advisoryPartitionSizeInBytes**) to prevent too many small tasks from being executed.	true
spark.sql.adaptive.coalescePartitions.initialPartitionNum	Initial number of shuffle partitions before merge. The default value is the same as the value of **spark.sql.shuffle.partitions**. This parameter is valid only when **spark.sql.adaptive.enabled** and **spark.sql.adaptive.coalescePartitions.enabled** are set to **true**. This parameter is optional. The initial number of partitions must be a positive number.	200
spark.sql.adaptive.coalescePartitions.minPartitionNum	Minimum number of shuffle partitions after merge. If this parameter is not set, the default degree of parallelism (DOP) of the Spark cluster is used. This parameter is valid only when **spark.sql.adaptive.enabled** and **spark.sql.adaptive.coalescePartitions.enable** are set to **true**. This parameter is optional. The initial number of partitions must be a positive number.	1

Parameter	Description	Default Value
spark.sql.adaptive.shuffle.targetPostShuffleInputSize	Target size of a partition after shuffling. Spark 3.0 and later versions do not support this parameter.	64MB
spark.sql.adaptive.advisoryPartitionSizeInBytes	Size of a shuffle partition (unit: byte) during adaptive optimization (**spark.sql.adaptive.enabled** is set to **true**). This parameter takes effect when Spark aggregates small shuffle partitions or splits shuffle partitions where skew occurs.	64MB
spark.sql.adaptive.fetchShuffleBlocksInBatch	Whether to obtain consecutive shuffle blocks in batches. For the same map job, reading consecutive shuffle blocks in batches can reduce I/Os and improve performance, instead of reading blocks one by one. Note that multiple consecutive blocks exist in a single read request only when **spark.sql.adaptive.enabled** and **spark.sql.adaptive.coalescePartitions.enabled** are set to **true**. This feature also relies on a relocatable serializer that uses cascading to support the codec and the latest version of the shuffle extraction protocol.	true
spark.sql.adaptive.localShuffleReader.enabled	If the value of this parameter is **true** and the value of **spark.sql.adaptive.enabled** is **true**, Spark attempts to use the local shuffle reader to read shuffle data when shuffling of partitions is not required, for example, after sort-merge join is converted to broadcast-hash join.	true
spark.sql.adaptive.skewJoin.enabled	Specifies whether to enable the function of automatic processing of the data skew in join operations. The function is enabled when this parameter is set to **true** and **spark.sql.adaptive.enabled** is set to **true**.	true

Parameter	Description	Default Value
spark.sql.adaptive.skewJoin.skewedPartitionFactor	This parameter is a multiplier used to determine whether a partition is a data skew partition. If the data size of a partition exceeds the value of this parameter multiplied by the median of the all partition sizes except this partition and exceeds the value of **spark.sql.adaptive.skewJoin.skewedPartitionThresholdInBytes**, this partition is considered as a data skew partition.	5
spark.sql.adaptive.skewJoin.skewedPartitionThresholdInBytes	If the partition size (unit: byte) is greater than the threshold as well as the product of the **spark.sql.adaptive.skewJoin.skewedPartitionFactor** value and the median partition size, skew occurs in the partition. Ideally, the value of this parameter should be greater than that of **spark.sql.adaptive.advisoryPartitionSizeInBytes.**.	256MB
spark.sql.adaptive.nonEmptyPartitionRatioForBroadcastJoin	If the ratio of non-null partitions is less than the value of this parameter when two tables are joined, broadcast hash join cannot be properly performed regardless of the partition size. This parameter is valid only when **spark.sql.adaptive.enabled** is set to **true**.	0.2

12.23.2.7.25 Configuring Event Log Rollover

Scenario

When the event log mode is enabled for Spark, that is, **spark.eventLog.enabled** is set to **true**, events are written to a configured log file to record the program running process. If a program, for example JDBCServer or Spark Streaming, runs for a long period of time and has run many jobs and tasks during this period, many events are recorded in the log file, significantly increasing the file size.

When log rollover is enabled, metadata events are written into the log file and job events are written into a new log file (whether a job event is written to the new log file depends on the file size). Metadata events include EnviromentUpdate, BlockManagerAdded, BlockManagerRemoved, UnpersistRDD, ExecutorAdded, ExecutorRemoved, MetricsUpdate, ApplicationStart, ApplicationEnd, and LogStart. Job events include StageSubmitted, StageCompleted, TaskResubmit, TaskStart,

TaskEnd, TaskGettingResult, JobStart, and JobEnd. For Spark SQL applications, job events also include ExecutionStart and ExecutionEnd.

The UI for the HistoryServer service of Spark is obtained by reading and parsing these log files. The memory size is preset before the HistoryServer process starts. Therefore, when the size of log files is large, loading and parsing these files may cause problems such as insufficient memory and driver GC.

To load large log files in small memory mode, you need to enable log rollover for large applications. Generally, it is recommended that this function be enabled for long-running applications.

Parameters

Log in to FusionInsight Manager, choose **Cluster** > **Services** > **Spark2x** > **Configurations**, click **All Configurations**, and search for the following parameters.

Parameter	Description	Default Value
spark.eventLog.rolling.enabled	Whether to enable rollover for event log files. If this parameter is set to **true**, the size of each event log file is reduced to the configured size.	true
spark.eventLog.rolling.maxFileSize	Maximum size of the event log file to be rolled over when **spark.eventlog.rolling.enabled** is set to **true**.	128M
spark.eventLog.compression.codec	Codec used to compress event logs. By default, Spark provides four types of codecs: LZ4, LZF, Snappy, and ZSTD. If this parameter is not specified, **spark.io.compression.codec** is used.	None
spark.eventLog.logStageExecutorMetrics	Whether to write each stage peak value (for each executor) of executor metrics to the event log.	false

12.23.2.8 Adapting to the Third-party JDK When Ranger Is Used

Scenarios

When Ranger is used as the permission management service of Spark SQL, the certificate in the cluster is required for accessing RangerAdmin. If you use a third-party JDK instead of the JDK or JRE in the cluster, RangerAdmin fails to be accessed. As a result, the Spark application fails to be started.

In this scenario, you need to perform the following operations to import the certificate in the cluster to the third-party JDK or JRE.

Configuration Method

Step 1 Run the following command to export the certificate from the cluster:

1. Install the cluster client. Assume that the installation path is **/opt/client**.
2. Run the following command to go to the client installation directory.

   cd /opt/client
3. Run the following command to configure environment variables:

   source bigdata_env
4. Generate the certificate file.

   keytool -export -alias fusioninsightsubroot -storepass changeit -keystore /opt/client/JRE/jre/lib/security/cacerts -file fusioninsightsubroot.crt

Step 2 Import the certificate in the cluster to the third-party JDK or JRE.

Copy the **fusioninsightsubroot.crt** file generated in **Step 1** to the third-party JRE node, set the **JAVA_HOME** environment variable of the node, and run the following command to import the certificate:

keytool -import -trustcacerts -alias fusioninsightsubroot -storepass changeit -file fusioninsightsubroot.crt -keystore MY_JRE/lib/security/cacerts

NOTE

MY_JRE indicates the installation path of the third-party JRE. Change it based on the site requirements.

----End

12.23.3 Spark2x Logs

Log Description

Log paths:

- Executor run log: **${BIGDATA_DATA_HOME}/hadoop/data${i}/nm/containerlogs/application_${appid}/container_{$contid}**

  NOTE

  The logs of running tasks are stored in the preceding path. After the running is complete, the system determines whether to aggregate the logs to an HDFS directory based on the Yarn configuration. For details, see **Common YARN Parameters**.
- Other logs: **/var/log/Bigdata/spark2x**

Log archiving rule:

- When tasks are submitted in **yarn-client** or **yarn-cluster** mode, executor log files are stored each time when the size of the log files reaches 50 MB. A maximum of 10 log files can be reserved without being compressed.
- The JobHistory2x log file is backed up each time when the size of the log file reaches 100 MB. A maximum of 100 log files can be reserved without being compressed.

- The JDBCServer2x log file is backed up each time when the size of the log file reaches 100 MB. A maximum of 100 log files can be reserved without being compressed.
- The IndexServer2x log file is backed up each time when the size of the log file reaches 100 MB. A maximum of 100 log files can be reserved without being compressed.
- The JDBCServer2x audit log file is backed up each time when the size of the log file reaches 20 MB by default. A maximum of 20 log files can be reserved without being compressed.
- The log file size and the number of compressed files to be reserved can be configured on FusionInsight Manager.

Table 12-406 Spark2x log list

Log Type	Name	Description
SparkResource2x logs	spark.log	Spark2x service initialization log
	prestart.log	Prestart script log
	cleanup.log	Cleanup log file for instance installation and uninstallation
	spark-availability-check.log	Spark2x service health check log
	spark-service-check.log	Spark2x service check log
JDBCServer2x logs	JDBCServer-start.log	JDBCServer2x startup log
	JDBCServer-stop.log	JDBCServer2x stop log
	JDBCServer.log	JDBCServer2x run log on the server
	jdbc-state-check.log	JDBCServer2x health check log
	jdbcserver-omm-pid***-gc.log.*.current	IJDBCServer2x process GC log
	spark-omm-org.apache.spark.sql.hive.thriftserver.HiveThriftProxyServer2-***.out*	JDBCServer2x process startup log. If the process stops, the **jstack** information is printed.
JobHistory2x logs	jobHistory-start.log	JobHistory2x startup log
	jobHistory-stop.log	JobHistory2x stop log
	JobHistory.log	JobHistory2x running process log
	jobhistory-omm-pid***-gc.log.*.current	JobHistory2x process GC log

Log Type	Name	Description
	spark-omm-org.apache.spark.deploy.history.HistoryServer-***.out*	JobHistory2x process startup log. If the process stops, the **jstack** information is printed.
IndexServer2x logs	IndexServer-start.log	IndexServer2x startup log
	IndexServer-stop.log	IndexServer2x stop log
	IndexServer.log	IndexServer2x run log on the server
	indexserver-state-check.log	IndexServer2x health check log
	indexserver-omm-pid***-gc.log.*.current	IndexServer2x process GC log
	spark-omm-org.apache.spark.sql.hive.thriftserver.IndexServerProxy-***.out*	IndexServer2x process startup log. If the process stops, the **jstack** information is printed.
Audit Log	jdbcserver-audit.log ranger-audit.log	JDBCServer2x audit log

Log levels

Table 12-407 describes the log levels supported by Spark2x. The priorities of log levels are ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-407 Log levels

Level	Description
ERROR	Error information about the current event processing
WARN	Exception information about the current event processing
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

NOTE

By default, the service does not need to be restarted after the Spark2x log levels are configured.

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Service** > **Spark2x** > **Configuration**.

Step 3 Select **All Configurations**.

Step 4 On the menu bar on the left, select the log menu of the target role.

Step 5 Select a desired log level.

Step 6 Click **Save**. Then, click **OK**.

----End

Log Format

Table 12-408 Log Format

Type	Format	Example
Run log	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2014-09-22 11:16:23,980 INFO DAGScheduler: Final stage: Stage 0(reduce at SparkPi.scala:35)

12.23.4 Obtaining Container Logs of a Running Spark Application

Container logs of running Spark applications are distributed on multiple nodes. This section describes how to quickly obtain container logs.

Scenario Description

You can run the **yarn logs** command to obtain the logs of applications running on Yarn. In different scenarios, you can run the following commands to obtain required logs:

1. Obtain complete logs of the application: **yarn logs --applicationId** *<appId>* **-out** *<outputDir>*.

   Example: **yarn logs --applicationId application_1574856994802_0016 -out /opt/test**

   The following figure shows the command output.

   a. If the application is running, container logs in the **dead** state cannot be obtained.

b. If the application is stopped, all archived container logs can be obtained.

2. Obtain logs of a specified container: **yarn logs -applicationId** *<appId>* **-containerId** *<containerId>*.

   Example: **yarn logs -applicationId application_1574856994802_0018 -containerId container_e01_1574856994802_0018_01_000003**

   The following figure shows the command output.

   a. If the application is running, container logs in the **dead** state cannot be obtained.

   b. If the application is stopped, you can obtain logs of any container.

3. Obtain container logs in any state: **yarn logs -applicationId** *<appId>* **-containerId** *<containerId>* **-nodeAddress** *<nodeAddress>*

   Example: **yarn logs -applicationId application_1574856994802_0019 -containerId container_e01_1574856994802_0019_01_000003 -nodeAddress 192-168-1-1:**8041

   Execution result: Logs of any container can be obtained.

   NOTE

   You need to set *nodeAddress* in the command. You can run the following command to obtain the value:

   yarn node -list -all

12.23.5 Small File Combination Tools

Tool Overview

In a large-scale Hadoop production cluster, HDFS metadata is stored in the NameNode memory, and the cluster scale is restricted by the memory limitation of each NameNode. If there are a large number of small files in the HDFS, a large amount of NameNode memory is consumed, which greatly reduces the read and write performance and prolongs the job running time. Based on the preceding information, the small file problem is a key factor that restricts the expansion of the Hadoop cluster.

This tool provides the following functions:

1. Checks the number of small files whose size is less than the threshold configured by the user in tables and returns the average size of all data files in the table directory.
2. Provides the function of combination table files. Users can set the average file size after combination.

Supported Table Types

Spark: Parquet, ORC, CSV, Text, and Json.

Hive: Parquet, ORC, CSV, Text, RCFile, Sequence and Bucket.

NOTE

1. After tables with compressed data are merged, Spark uses the default compression format Snappy for data compression. You can configure **spark.sql.parquet.compression.codec** (available values: **uncompressed**, **gzip**, **lzo**, and **snappy**) and **spark.sql.orc.compression.codec** (available values: **uncompressed**, **zlib**, **lzo**, and **snappy**) on the client to select the compression format for the Parquet and ORC tables. Compression formats available for Hive and Spark tables are different, except the preceding compression formats, other compression formats are not supported.
2. To merge bucket table data, you need to add the following configurations to the **hive-site.xml** file on the Spark2x client:

```
<property>
<name>hive.enforce.bucketing</name>
<value>false</value>
</property>
<property>
<name>hive.enforce.sorting</name>
<value>false</value>
</property>
```

3. Spark does not support the feature of encrypting data columns in Hive.

Tool Usage

Download and install the client. For example, the installation directory is **/opt/client**. Go to **/opt/client/Spark2x/spark/bin** and run the **mergetool.sh** script.

Environment variables loading

source /opt/client/bigdata_env

source /opt/client/Spark2x/component_env

Scanning function

Command: **sh mergetool.sh scan <db.table> <filesize>**

The format of *db.table* is *Database name*,*Table name*. *filesize* is the user-defined threshold of the small file size (unit: MB). The returned result is the number of files that is smaller than the threshold and the average size of data files in the table directory.

Example: **sh mergetool.sh scan default.table1 128**

Combination function

Command: **sh mergetool.sh merge <db.table> <filesize> <shuffle>**

The format of *db.table* is *Database name,Table name*. **filesize** is the user-defined average file size after file combination (unit: MB). **shuffle** is a Boolean value, and the value is **true** or **false**, which is used to configure whether to allow data to be shuffled during the merge.

Example: **sh mergetool.sh merge default.table1 128 false**

If the following information is displayed, the operation is successful:

```
SUCCESS: Merge succeeded
```

NOTE

1. Ensure that the current user is the owner of the merged table.
2. Before combination, ensure that HDFS has sufficient storage space, greater than the size of the combined table.
3. Table data must be combined separately. If a table is read during table data combination, the file may not be found temporarily. After the combination is complete, this problem is resolved. During the combination, do not write data to the corresponding tables. Otherwise, data inconsistency may occur.
4. If an error occurs indicating that the file does not exist when the query of data in a partitioned table is performed on the session spark-beeline/spark-sql that is always in the connected status. You can run the **refresh table***Table name* command as prompted to query the data again.
5. Configure **filesize** based on the site requirements. For example, you can set **filesize** to a value greater than the average during file merging after obtaining the average file size by file scan. Otherwise, the number of files may increase after the file merging.
6. During the file merging, data in the original tables is removed to the recycle bin. In the case of any exception occurs on the data after file merging, the data in the original tables is used to replace the damaged data. If an exception occurs during the process, restore the data in the trash directory by using the **mv** command in HDFS.
7. In the HDFS router federation scenario, if the target NameService of the table root path is different from that of the root path **/user**, you need to manually clear the original table files stored in the recycle bin during the second combination. Otherwise, the combination fails.
8. This tool uses the configuration of the client. Performance optimization can be performed modifying required configuration in the client configuration file.

shuffle configuration

For the combination function, you can roughly estimate the change on the number of partitions before and after the combination.

Generally, if the number of old partitions is greater than the number of new partitions, set **shuffle** to **false**. However, if the number of old partitions is much greater than that of new partitions (for example, more than 100 times), you can set **shuffle** to **true** to increase the degree of parallelism and improve the combination speed.

NOTICE

- If **shuffle** is set to **true** (repartition), the performance is improved. However, due to the particularity of the Parquet and ORC storage modes, repartition will reduce the compression ratio and the total size of the table in HDFS increases by 1.3 times.
- If **shuffle** is set to **false** (coalesce), the merged files may have some difference in size, which is close to the value of the configured **filesize**.

Log storage location

The default log storage path is **/tmp/SmallFilesLog.log4j**. To customize the log storage path, you can configure **log4j.appender.logfile.File** in **/opt/client/Spark2x/spark/tool/log4j.properties**.

12.23.6 Using CarbonData for First Query

Tool Overview

The first query of CarbonData is slow, which may cause a delay for nodes that have high requirements on real-time performance.

The tool provides the following functions:

- Preheat the tables that have high requirements on query delay for the first time.

Tool Usage

Download and install the client. For example, the installation directory is **/opt/client**. Go to the **/opt/client/Spark2x/spark/bin** directory and run **start-prequery.sh**.

Configure **prequeryParams.properties** by referring to **Table 12-409**.

Table 12-409 Parameters

Parameter	Description	Example
spark.prequery.period.max.minute	Maximum preheating duration, in minutes.	60
spark.prequery.tables	Table name configuration, *database.table:int*. The table name supports the wildcard (*). **int** indicates the duration (unit: day) within which the table is updated before it is preheated.	default.test*:10
spark.prequery.maxThreads	Maximum number of concurrent threads during preheating	50
spark.prequery.sslEnable	The value is **true** in security mode and **false** in non-security mode.	true
spark.prequery.driver	IP address and port number of JDBCServer. The format is *IP address:Port number*. If multiple servers need to be preheated, enter multiple *IP address:Port number* of the servers and separate them with commas (,).	192.168.0.2:22550

Parameter	Description	Example
spark.prequery.sql	SQL statement for preheating. Different statements are separated by colons (:).	SELECT COUNT(*) FROM %s;SELECT * FROM %s LIMIT 1
spark.security.url	URL required by JDBC in security mode	;saslQop=auth-conf;auth=KERBEROS;principal=spark2x/hadoop.hadoop.com@HADOOP.COM;

NOTE

The statement configured in **spark.prequery.sql** is executed in each preheated table. The table name is replaced with **%s**.

Script Usage

Command format: **sh start-prequery.sh**

To run this command, place **user.keytab** or **jaas.conf** (either of them) and **krb5.conf** (mandatory) in the **conf** directory.

NOTE

- Currently, this tool supports only Carbon tables.
- This tool initializes the Carbon environment and pre-reads table metadata to JDBCServer. Therefore, this tool is more suitable for multi-active instances and static allocation mode.

12.23.7 Spark2x Performance Tuning

12.23.7.1 Spark Core Tuning

12.23.7.1.1 Data Serialization

Scenario

Spark supports the following types of serialization:

- JavaSerializer
- KryoSerializer

Data serialization affects the Spark application performance. In specific data format, KryoSerializer offers 10X higher performance than JavaSerializer. For Int data, performance optimization can be ignored.

KryoSerializer depends on Chill of Twitter. Not all Java Serializable objects support KryoSerializer. Therefore, class must be manually registered.

Serialization involves task serialization and data serialization. Only JavaSerializer can be used for Spark task serialization. JavaSerializer and KryoSerializer can be used for data serialization.

Procedure

When the Spark program is running, a large volume of data needs to be serialized during the shuffle and RDD cache procedures. By default, JavaSerializer is used. You can also configure KryoSerializer as the data serializer to improve serialization performance.

Add the following code to enable KryoSerializer to be used:

- Implement the class registrar and manually register the class.

```
package com.etl.common;

import com.esotericsoftware.kryo.Kryo;
import org.apache.spark.serializer.KryoRegistrator;

public class DemoRegistrator implements KryoRegistrator
{
   @Override
   public void registerClasses(Kryo kryo)
   {
      //Class examples are given below. Register the custom classes.
      kryo.register(AggrateKey.class);
      kryo.register(AggrateValue.class);
   }
}
```

  You can configure **spark.kryo.registrationRequired** on Spark client. Whether to require registration with Kryo. If set to 'true', Kryo will throw an exception if an unregistered class is serialized. If set to false (the default), Kryo will write unregistered class names along with each object. Writing class names can cause significant performance overhead. This operation will affect the system performance. If the value of **spark.kryo.registrationRequired** is configured to **true**, you need to manually register the class. For a class that is not serialized, the system will not automatically write the class name, but display an exception. Compare the configuration of **true** with that of **false**, the configuration of **true** has the better performance.

- Configure KryoSerializer as the data serializer and class registrar.

```
val conf = new SparkConf()
conf.set("spark.serializer", "org.apache.spark.serializer.KryoSerializer")
.set("spark.kryo.registrator", "com.etl.common.DemoRegistrator")
```

12.23.7.1.2 Optimizing Memory Configuration

Scenario

Spark is a memory-based computing frame. If the memory is insufficient during computing, the Spark execution efficiency will be adversely affected. You can determine whether memory becomes the performance bottleneck by monitoring garbage collection (GC) and evaluating the resilient distributed dataset (RDD) size in the memory, and take performance optimization measures.

To monitor GC of node processes, add the -verbose:gc -XX:+PrintGCDetails -XX:+PrintGCTimeStamps parameter to the **spark.driver.extraJavaOptions** and **spark.executor.extraJavaOptions** in the client configuration file **conf/spark-default.conf**. If "Full GC" is frequently reported, GC needs to be optimized. Cache

the RDD and query the RDD size in the log. If a large value is found, change the RDD storage level.

Procedure

- To optimize GC, adjust the ratio of the young generation and tenured generation. Add **-XX:NewRatio** parameter to the **spark.driver.extraJavaOptions** and **spark.executor.extraJavaOptions** in the client configuration file **conf/spark-default.conf**. For example, export SPARK_JAVA_OPTS=" -XX:NewRatio=2". The new generation accounts for 1/3 of the heap, and the tenured generation accounts for 2/3.
- Optimize the RDD data structure when compiling Spark programs.
  - Use primitive arrays to replace fastutil arrays, for example, use fastutil library.
  - Avoid nested structure.
  - Avoid using String in keys.
- Suggest serializing the RDDs when developing Spark programs.

  By default, data is not serialized when RDDs are cached. You can set the storage level to serialize the RDDs and minimize memory usage. For example:

  ```
  testRDD.persist(StorageLevel.MEMORY_ONLY_SER)
  ```

12.23.7.1.3 Setting the DOP

Scenario

The degree of parallelism (DOP) specifies the number of tasks to be executed concurrently. It determines the number of data blocks after the shuffle operation. Configure the DOP to improve the processing capability of the system.

Query the CPU and memory usage. If the tasks and data are not evenly distributed among nodes, increase the DOP. Generally, set the DOP to two or three times that of the total CPUs in the cluster.

Procedure

Configure the DOP parameter using one of the following methods based on the actual memory, CPU, data, and application logic conditions:

- Configure the DOP parameter in the operation function that generates the shuffle. This method has the highest priority.

  ```
  testRDD.groupByKey(24)
  ```

- Configure the DOP using **spark.default.parallelism**. This method has the lower priority than the preceding one.

  ```
  val conf = new SparkConf();
  conf.set("spark.default.parallelism", 24);
  ```

- Configure the value of **spark.default.parallelism** in the **$SPARK_HOME/conf/spark-defaults.conf** file. This method has the lowest priority.

  ```
  spark.default.parallelism    24
  ```

12.23.7.1.4 Using Broadcast Variables

Scenario

Broadcast distributes data sets to each node. It allows data to be obtained locally when a data set is needed during a Spark task. If broadcast is not used, data serialization will be scheduled to tasks each time when a task requires data sets. It is time-consuming and makes the task get bigger.

1. If a data set will be used by each slice of a task, broadcast the data set to each node.
2. When small and big tables need to be joined, broadcast small tables to each node. This eliminates the shuffle operation, changing the join operation into a common operation.

Procedure

Add the following code to broadcast the testArr data to each node:

```
def main(args: Array[String) {
  ...
  val testArr: Array[Long] = new Array[Long](200)
  val testBroadcast: Broadcast[Array[Long]] = sc.broadcast(testArr)
  val resultRdd: RDD[Long] = inpputRdd.map(input => handleData(testBroadcast, input))
  ...
}

def handleData(broadcast: Broadcast[Array[Long]], input: String) {
  val value = broadcast.value
  ...
}
```

12.23.7.1.5 Using the external shuffle service to improve performance

Scenario

When the Spark system runs applications that contain a shuffle process, an executor process also writes shuffle data and provides shuffle data for other executors in addition to running tasks. If the executor is heavily loaded and GC is triggered, the executor cannot provide shuffle data for other executors, affecting task running.

The external shuffle service is an auxiliary service in NodeManager. It captures shuffle data to reduce the load on executors. If GC occurs on an executor, tasks on other executors are not affected.

Procedure

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Spark2x** > **Configurations**. Select **All Configurations**.

Step 3 Choose **SparkResource2x** > **Default** and modify the following parameters.

Table 12-410 Parameter list

Parameter	Default Value	Changed To
spark.shuffle.service.enabled	false	true

Step 4 Restart the Spark2x service for the configuration to take effect.

NOTE

To use the External Shuffle Service function on the Spark2x client, you need to download and install the Spark2x client again.

----End

12.23.7.1.6 Configuring Dynamic Resource Scheduling in Yarn Mode

Scenario

Resources are a key factor that affects Spark execution efficiency. When a long-running service (such as the JDBCServer) is allocated with multiple executors without tasks but resources of other applications are insufficient, resources are wasted and scheduled improperly.

Dynamic resource scheduling can add or remove executors of applications in real time based on the task load. In this way, resources are dynamically scheduled to applications.

Procedure

Step 1 Configure the external shuffle service.

Step 2 Log in to FusionInsight Manager, and choose **Cluster** > *Name of the desired cluster* > **Service** > **Spark2x** > **Configuration** > **All Configurations**. Enter **spark.dynamicAllocation.enabled** in the search box and set its value to **true** to enable the dynamic resource scheduling function. This function is disabled by default.

----End

Table 12-411 lists some optional configuration items.

Table 12-411 Parameters for dynamic resource scheduling

Configuration Item	Description	Default Value
spark.dynamicAllocation.minExecutors	Indicates the minimum number of executors.	0
spark.dynamicAllocation.initialExecutors	Indicates the number of initial executors.	0
spark.dynamicAllocation.maxExecutors	Indicates the maximum number of executors.	2048

Configuration Item	Description	Default Value
spark.dynamicAllocation.schedulerBacklogTimeout	Indicates the first timeout period for scheduling.	1s
spark.dynamicAllocation.sustainedSchedulerBacklogTimeout	Indicates the second and later timeout interval for scheduling.	1s
spark.dynamicAllocation.executorIdleTimeout	Indicates the idle timeout interval for common executors.	60s
spark.dynamicAllocation.cachedExecutorIdleTimeout	Indicates the idle timeout interval for executors with cached blocks.	• JDBCServer2x: 2147483647s • IndexServer2x: 2147483647s • SparkResource2x: 120

NOTE

The external shuffle service must be configured before using the dynamic resource scheduling function.

12.23.7.1.7 Configuring Process Parameters

Scenario

There are three processes in Spark on Yarn mode: driver, ApplicationMaster, and executor. The Driver and Executor handle the scheduling and running of the task. The ApplicationMaster handles the start and stop of the container.

Therefore, the configuration of the driver and executor is very important to run the Spark application. You can optimize the performance of the Spark cluster according to the following procedure.

Procedure

Step 1 Configure the driver memory.

The driver schedules tasks and communicates with the executor and the ApplicationMaster. Add driver memory when the number and parallelism level of the tasks increases.

You can configure the driver memory based on the number of the tasks.

- Set **spark.driver.memory** in **spark-defaults.conf** to a proper value.
- Add the **--driver-memory MEM** parameter to configure the memory when using the **spark-submit** command.

Step 2 Configure the number of the executors.

One core in an executor can run one task at the same time. Therefore, more tasks can be processed at the same time if you increase the number of the executors. You can add the number of the executors to increase the efficiency if resources are sufficient.

- Set **spark.executor.instance** in **spark-defaults.conf** or **SPARK_EXECUTOR_INSTANCES** in **spark-env.sh** to a proper value.
- Add the **--num-executors NUM** parameter to configure the number of the executors when using the **spark-submit** command.

Step 3 Configure the number of the executor cores.

Multiple cores in an executor can run multiple tasks at the same time, which increases the task concurrency. However, because all cores share the memory of an executor, you need to balance the memory and the number of cores.

- Set **spark.executor.cores** in **spark-defaults.conf** or **SPARK_EXECUTOR_CORES** in **spark-env.sh** to a proper value.
- When you run the **spark-submit** command, add the **--executor-cores NUM** parameter to set the number of executor cores.

Step 4 Configure the executor memory.

The executor memory is used for task execution and communication. You can increase the memory for a big task that needs more resources, and reduce the memory to increase the concurrency level for a small task that runs fast.

- Set **spark.executor.memory** in **spark-defaults.conf** or **SPARK_EXECUTOR_MEMORY** in **spark-env.sh** to a proper value.
- When you run the **spark-submit** command, add the **--executor-memory MEM** parameter to set the memory.

----End

Example

- During the **spark wordcount** calculation, the amount of data is 1.6 TB and the number of the executors is 250.

  The execution fails under the default configuration, and the **Futures timed out** and **OOM** errors occur.

  However each task of wordcount is small and runs fast, the amount of the data is big and the tasks are too many. Therefore the objects on the driver end become huge when there are many tasks. Besides the fact that the executor communicates with the driver once each task is finished, the problem of disconnection between processes caused by insufficient memory occurs.

  The application runs successfully when the memory of the Driver is set to 4 GB.

- Many errors still occurred in the default configuration when running TPC-DS test on JDBCServer, such as "Executor Lost". When there is 30 GB of driver memory, 2 executor cores, 125 executors, and 6 GB of executor memory, all tasks can be successfully executed.

12.23.7.1.8 Designing the Direction Acyclic Graph (DAG)

Scenario

Optimal program structure helps increase execution efficiency. During application programming, avoid shuffle operations and combine narrow-dependency operations.

Procedure

This topic describes how to design the DAG using the following example:

- **Data format**: Time when a vehicle passes a toll station, license plate number, toll station number, and more
- **Logic**: Two vehicles are determined to be traveling together if the following conditions are met:
  - Both vehicles pass the same toll stations in the same sequence.
  - The difference between the time that the vehicles pass the same toll station is smaller than a specified value.

There are two implementation ways for this example. **Figure 12-56** shows the logic of implementation 1 and **Figure 12-57** shows logic of implementation 2.

Figure 12-56 Implementation logic 1

Logic description:

1. Collect information about the toll stations passed by each vehicle based on the vehicle license plate number and sort the toll stations.

   The following data is obtained: vehicle license plate number 1, [(time, toll station 3), (time, toll station 2), (time, toll station 4), (time, toll station 5)]

2. Determine the sequence in which the vehicle passed through.

   (toll station 3, (vehicle license plate number 1, time, 1st toll station))

   (toll station 2, (vehicle license plate number 1, time, 2nd toll station))

   (toll station 4, (vehicle license plate number 1, time, 3rd toll station))

   (toll station 5, (vehicle license plate number 1, time, 4th toll station))

3. Aggregate data by toll station.

   toll station 1, [(vehicle license plate number 1, time, 1st toll station), (vehicle license plate number 2, time, 5th toll station), (vehicle license plate number 3, time, 2nd toll station)]

4. Determine whether the time difference that two vehicles passed through the same toll station is below the specified value. If yes, fetch information about the two vehicles.

   (vehicle license plate number 1, vehicle license plate number 2),(1st toll station, 5th toll station)

   (vehicle license plate number 1, vehicle license plate number 3),(1st toll station, 2nd toll station)

5. Aggregate data based on the vehicle license plate numbers that passed through the same toll stations.

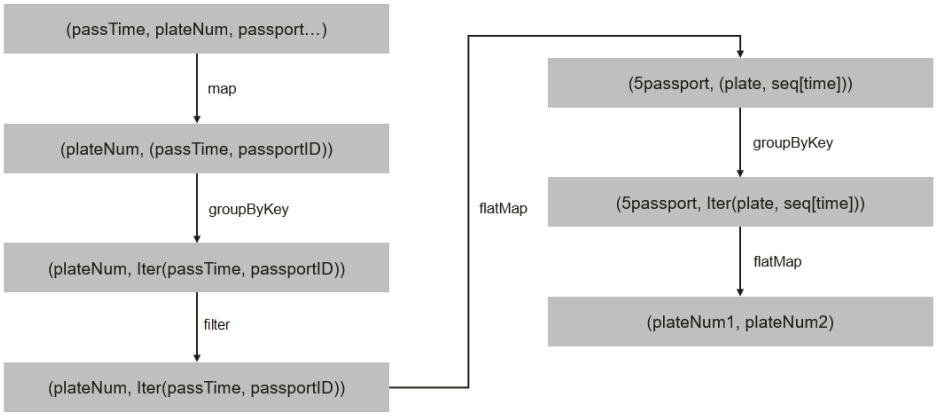
   (vehicle license plate number 1, vehicle license plate number 2), [(1st toll station, 5th toll station), (2nd toll station, 6th toll station), (1st toll station, 7th toll station), (3rd toll station, 8th toll station)]

6. If the two vehicles pass through the same toll stations in sequence, for example, toll stations 3, 4, 5 are the first, second, and third toll station passed by vehicle 1 and the 6th, 7th, and 8th toll station passed by vehicle 2, and the number of toll stations meets the specified requirements, the two vehicles are determined to be traveling together.

The logic of implementation 1 has the following disadvantages:

- The logic is complex.
- Too many shuffle operations affect performance.

Figure 12-57 Implementation logic 2



Logic description:

1. Collect information about the toll stations passed by each vehicle based on the vehicle license plate number and sort the toll stations.

   The following data is obtained: vehicle license plate number 1, [(time, toll station 3), (time, toll station 2), (time, toll station 4), (time, toll station 5)]

2. Based on the number of toll stations (the number is 3 in this example) that must be passed by these vehicles, divide the toll station sequence as follows:

toll station 3 > toll station 2 > toll station 4, (vehicle license plate number 1, [time passing through toll station 3, time passing through toll station 2, time passing through toll station 4])

toll station 2 > toll station 4 > toll station 5, (vehicle license plate number 1, [time passing through toll station 2, time passing through toll station 4, time passing through toll station 5])

3. Aggregate information about vehicles that pass the same toll stations in the same sequence.

   toll station 3 > toll station 2 > toll station 4, [(vehicle license plate number 1, [time passing through toll station 3, time passing through toll station 2, time passing through toll station 4]), (vehicle license plate number 2, [time passing through toll station 3, time passing through toll station 2, time passing through toll station 4]), (vehicle license plate number 3, [time passing through toll station 3, time passing through toll station 2, time passing through toll station 4])]

4. Determine whether the time difference that these vehicles passed through the same toll station is below the specified value. If yes, the vehicles are determined to be traveling together.

The logic of implementation 2 has the following advantages:

- The logic is simplified.
- One **groupByKey** is reduced, that is, one less shuffle operation is performed. It helps improve performance.

12.23.7.1.9 Experience

Use mapPartitions to calculate data by partition.

If the overhead of each record is high, for example:

```
rdd.map{x=>conn=getDBConn;conn.write(x.toString);conn.close}
```

Use mapPartitions to calculate data by partition.

```
rdd.mapPartitions(records => conn.getDBConn;for(item <- records)
write(item.toString); conn.close)
```

Use mapPartitions to flexibly operate data. For example, to calculate the TopN of a large data, mapPartitions can be used to calculate the TopN of each partition and then sort the TopN of all partitions when N is small. Compared with sorting full data for the TopN, this method has the higher efficiency.

Use coalesce to adjust the number of slices.

Use coalesce to adjust the number of slices. There are two coalesce functions:

```
coalesce(numPartitions: Int, shuffle: Boolean = false)
```

When **shuffle** is set to **true**, the function is the same as repartition(numPartitions:Int). Partitions are recreated using the shuffle. When **shuffle** is set to **false**, partitions of the parent resilient distributed datasets (RDD) are calculated in the same task. In this case, if the value of **numPartitions** is larger than the number of sections of the parent RDD, partitions will not be recreated.

The following scenario is encountered, you can choose the coalesce operator:

- If the previous operation involves a large number of filters, use coalesce to minimize the number of zero-loaded tasks. In coalesce(numPartitions, false), the value of **numPartitions** is smaller than the number of sections of the parent RDD.
- Use coalesce when the number of slices entered is too big to execute.
- Use coalesce when the programs are suspended in the shuffle operation because of a large number of tasks or the Linux resources are limited. In this case, use coalesce(numPartitions, true) to recreate partitions.

Configure a localDir for each disk.

During the shuffle procedure of Spark, data needs to be written into local disks. The performance bottleneck of Spark is shuffle, and the bottleneck of shuffle is the I/O. To improve the I/O performance, you can configure multiple disks to implement concurrent data writing. If a node is mounted with multiple disks, configure a Spark local Dir for each disk. This can effectively distribute shuffle files in multiple locations, improving disk I/O efficiency. The performance cannot be improved effectively if a disk is configured with multiple directories.

Collect small data sets.

The collect operation does not apply to a large data volume.

When the collect operation is performed, the Executor data will be sent to the Driver. Before performing this operation, ensure that the memory of Driver is sufficient. Otherwise, the Driver process may encounter an OutOfMemory error. If the data volume is unknown, perform the saveAsTextFile operation to write data into the HDFS. If the data volume is known and the Driver has sufficient memory, perform the collect operation.

Use reduceByKey

reduceByKey causes local aggregation on the Map side, which offers a smooth shuffle procedure. The shuffle operations, like groupByKey, will not perform aggregation on the Map side. Therefore, use reduceByKey as possible as you can, and avoid groupByKey().map(x=>(x._1,x._2.size)).

Broadcast map instead of array.

If table query is required for each record of the data transmitted from the Driver side, broadcast the data in the set/map instead of Iterator. The query speed of Set/Map is approximately O(1), while the query speed of Iterator is O(n).

Avoid data skew.

If data skew occurs (certain data volume is extremely large), the execution time of tasks is inconsistent even if there is no Garbage Collection (GC).

- Redefine the keys. Use keys of smaller granularity to optimize the task size.
- Modify the degree of parallelism (DOP).

Optimize the data structure.

- Store data by column. As a result, only the required columns are scanned when data is read.
- When using the Hash Shuffle, set **spark.shuffle.consolidateFiles** to **true** to combine the intermediate files of shuffle, minimize the number of shuffle files and file I/O operations, and improve performance. The number of final files is the number of reduce tasks.

12.23.7.2 Spark SQL and DataFrame Tuning

12.23.7.2.1 Optimizing the Spark SQL Join Operation

Scenario

When two tables are joined in Spark SQL, the broadcast function (see section "Using Broadcast Variables") can be used to broadcast tables to each node. This minimizes shuffle operations and improves task execution efficiency.

NOTE

The join operation refers to the inner join operation only.

Procedure

The following describes how to optimize the join operation in Spark SQL. Assume that both tables A and B have the **name** column. Join tables A and B as follows:

1. Estimate the table sizes.

   Estimate the table size based on the size of data loaded each time.

   You can also check the table size in the directory of the Hive database. In the **hive-site.xml** configuration file of Spark, view the Hive database directory, which is **/user/hive/warehouse** by default. The default Hive database directory for multi-instance Spark is **/user/*hive*/warehouse**, for example, **/user/hive1/warehouse**.

   ```
   <property>
     <name>hive.metastore.warehouse.dir</name>
    <value>${test.warehouse.dir}</value>
    <description></description>
   </property>
   ```

   Run the **hadoop** command to check the size of the table. For example, run the following command to view the size of table **A**:

   ```
   hadoop fs -du -s -h ${test.warehouse.dir}/a
   ```

   NOTE

   To perform the broadcast operation, ensure that at least one table is not empty.

2. Configure a threshold for automatic broadcast.

   The threshold for triggering broadcast for a table is 10485760 (that is, 10 MB) in Spark. If either of the table sizes is smaller than 10 MB, skip this step.

   Table 12-412 lists configuration parameters of the threshold for automatic broadcasting.

Table 12-412 Parameter description

Parameter	Default Value	Description
spark.sql.autoBroadcastJoinThreshold	10485760	Indicates the maximum value for the broadcast configuration when two tables are joined. • When the size of a field in a table involved in an SQL statement is less than the value of this parameter, the system broadcasts the SQL statement. • If the value is set to **-1**, broadcast is not performed. For details, visit **https://archive.apache.org/dist/spark/docs/3.1.1/sql-programming-guide.html**.

Methods for configuring the threshold for automatic broadcasting:

- Set **spark.sql.autoBroadcastJoinThreshold** in the **spark-defaults.conf** configuration file of Spark.
  ```
  spark.sql.autoBroadcastJoinThreshold = <size>
  ```
- Run the Hive command to set the threshold. Before joining the tables, run the following command:
  ```
  SET spark.sql.autoBroadcastJoinThreshold=<size>;
  ```

3. Join the tables.
   - The size of each table is smaller than the threshold.
     - If the size of table A is smaller than that of table B, run the following command:
       ```
       SELECT A.name FROM B JOIN A ON A.name = B.name;
       ```
     - If the size of table B is smaller than that of table A, run the following command:
       ```
       SELECT A.name FROM A JOIN B ON A.name = B.name;
       ```
   - One table size is smaller than the threshold, while the other table size is greater than the threshold.

     Broadcast the smaller table.
   - The size of each table is greater than the threshold.

     Compare the size of the field involved in the query with the threshold.
     - If the values of the fields in a table are smaller than the threshold, the corresponding data in the table is broadcast.
     - If the values of the fields in the two tables are greater than the threshold, do not broadcast either of the table.
4. (Optional) In the following scenarios, you need to run the Analyze command (***ANALYZE TABLE tableName COMPUTE STATISTICS noscan;***) to update metadata before performing the broadcast operation:

  - The table to be broadcasted is a newly created partitioned table and the file type is non-Parquet.
  - The table to be broadcasted is a newly updated partitioned table.

Reference

A task is ended if a timeout occurs during the execution of the to-be-broadcasted table.

By default, BroadCastJoin allows only 5 minutes for the to-be-broadcasted table calculation. If the time is exceeded, a timeout will occur. However, the broadcast task of the to-be-broadcasted table calculation is still being executed, resulting in resource waste.

The following methods can be used to address this issue:

- Modify the value of **spark.sql.broadcastTimeout** to increase the timeout duration.
- Reduce the value of **spark.sql.autoBroadcastJoinThreshold** to disable the optimization of BroadCastJoin.

12.23.7.2.2 Improving Spark SQL Calculation Performance Under Data Skew

Scenario

When multiple tables are joined in Spark SQL, skew occurs in join keys and the data volume in some Hash buckets is much higher than that in other buckets. As a result, some tasks with a large amount of data run slowly, resulting low computing performance. Other tasks with a small amount of data are quickly completed, which frees many CPUs and results in a waste of CPU resources.

If the automatic data skew function is enabled, data that exceeds the bucketing threshold is bucketed. Multiple tasks proceed data in one bucket. Therefore, CUP usage is enhanced and the system performance is improved.

NOTE

Data that has no skew is bucketed and run in the original way.

Restrictions:

- Only the join between two tables is supported.
- FULL OUTER JOIN data does not support data skew.

  For example, the following SQL statement indicates that the skew of table **a** or table **b** cannot trigger the optimization.

  select aid FROM a FULL OUTER JOIN b ON aid=bid;
- LEFT OUTER JOIN data does not support the data skew of the right table.

  For example, the following SQL statement indicates that the skew of table **b** cannot trigger the optimization.

  select aid FROM a LEFT OUTER JOIN b ON aid=bid;
- RIGHT OUTER JOIN does not support the data skew of the left table.

  For example, the following SQL statement indicates that the skew of table **a** cannot trigger the optimization.

select aid FROM a RIGHT OUTER JOIN b ON aid=bid;

Configuration Description

Add the following parameters in the following table to the **spark-defaults.conf** configuration file on the Spark driver.

Table 12-413 Parameter description

Parameter	Description	Default Value
spark.sql.adaptive.enabled	The switch to enable the adaptive execution feature. Note: If AQE and Static Partition Pruning (DPP) are enabled at the same time, DPP takes precedence over AQE during SparkSQL task execution. As a result, AQE does not take effect. The DPP in the cluster is enabled by default. Therefore, you need to disable it when enabling the AQE.	false
spark.sql.optimizer.dynamicPartitionPruning.enabled	The switch to enable DPP.	true
spark.sql.adaptive.skewJoin.enabled	Specifies whether to enable the function of automatic processing of the data skew in join operations. The function is enabled when this parameter is set to **true** and **spark.sql.adaptive.enabled** is set to **true**.	true
spark.sql.adaptive.skewJoin.skewedPartitionFactor	This parameter is a multiplier used to determine whether a partition is a data skew partition. If the data size of a partition exceeds the value of this parameter multiplied by the median of the all partition sizes except this partition and exceeds the value of **spark.sql.adaptive.skewJoin.skewedPartitionThresholdInBytes**, this partition is considered as a data skew partition.	5
spark.sql.adaptive.skewjoin.skewedPartitionThresholdInBytes	If the partition size (unit: byte) is greater than the threshold as well as the product of the **spark.sql.adaptive.skewJoin.skewedPartitionFactor** value and the median partition size, skew occurs in the partition. Ideally, the value of this parameter should be greater than that of **spark.sql.adaptive.advisoryPartitionSizeInBytes.**.	256MB
spark.sql.adaptive.shuffle.targetPostShuffleInputSize	Minimum amount of shuffle data processed by each task. The unit is byte.	67108864

12.23.7.2.3 Optimizing Spark SQL Performance in the Small File Scenario

Scenario

A Spark SQL table may have many small files (far smaller than an HDFS block), each of which maps to a partition on the Spark by default. In other words, each small file is a task. If the small files are great in number, Spark must initiate a large number of tasks. If shuffle operations exist in Spark SQL, the number of hash buckets increases, affecting performance.

In this scenario, you can manually specify the split size of each task to avoid an excessive number of tasks and improve performance.

NOTE

If the SQL logic does not involve shuffle operations, this optimization does not improve performance.

Configuration

If you want to enable small file optimization, configure the **spark-defaults.conf** file on the Spark client.

Table 12-414 Parameter description

Parameter	Description	Default Value
spark.sql.files.maxPartitionBytes	The maximum number of bytes that can be packed into a single partition when a file is read. Unit: byte	134217728 (128 MB)
spark.files.openCostInBytes	The estimated cost to open a file, measured by the number of bytes that can be scanned in the same time. This is used when putting multiple files into a partition. It is better to over estimate, then the partitions with small files will be faster than partitions with larger files.	4 MB

12.23.7.2.4 Optimizing the INSERT...SELECT Operation

Scenario

The INSERT...SELECT operation needs to be optimized if any of the following conditions is true:

- Many small files need to be queried.
- A few large files need to be queried.
- The INSERT...SELECT operation is performed by a non-spark user in Beeline/JDBCServer mode.

Procedure

Optimize the INSERT...SELECT operation as follows:

- If the table to be created is the Hive table, set the storage type to Parquet. This enables INSERT...SELECT statements to be run faster.
- Perform the INSERT...SELECT operation as a spark-sql user or spark user (if in Beeline/JDBCServer mode). In this way, it is no longer necessary to change the file owner repeatedly, accelerating the execution of INSERT...SELECT statements.

  NOTE

  In Beeline/JDBCServer mode, the executor user is the same as the driver user. The driver user is a spark user because the driver is a part of JDBCServer service and started by a spark user. If the Beeline user is not a spark user, the file owner must be changed to the Beeline user (actual user) because the executor is unaware of the Beeline user.
- If many small files need to be queried, set spark.sql.files.maxPartitionBytes and spark.files.openCostInBytes to set the maximum size in bytes of partition and combine multiple small files in a partition to reduce file amount. This accelerates file renaming, ultimately enabling INSERT...SELECT statements to be run faster.

NOTE

The preceding optimizations are not a one-size-fits-all solution. In the following scenario, it still takes long to perform the INSERT...SELECT operation:

The dynamic partitioned table contains many partitions.

12.23.7.2.5 Multiple JDBC Clients Concurrently Connecting to JDBCServer

Scenario

Multiple clients can be connected to JDBCServer at the same time. However, if the number of concurrent tasks is too large, the default configuration of JDBCServer must be optimized to adapt to the scenario.

Procedure

1. Set the fair scheduling policy of JDBCServer.

   The default scheduling policy of Spark is **FIFO**, which may cause a failure of short tasks in multi-task scenarios. Therefore, the fair scheduling policy must be used in multi-task scenarios to prevent task failure.

   a. For details about how to configure Fair Scheduler in Spark, visit **http://archive.apache.org/dist/spark/docs/3.1.1/job-scheduling.html#scheduling-within-an-application**.

   b. Configure Fair Scheduler on the JDBC client.

      i. In the Beeline command line client or the code defined by JDBC, run the following statement:

      PoolName is a scheduling pool for Fair Scheduler.

      ```
      SET spark.sql.thriftserver.scheduler.pool=PoolName;
      ```

ii. Run the SQL command. The Spark task will be executed in the preceding scheduling pool.

2. Set the **BroadCastHashJoin** timeout interval.

   There is a timeout parameter of **BroadCastHashJoin**. The task query fails if the query period exceeds the preset timeout interval. In multi-task scenarios, the Spark task of BroadCastHashJoin may fail due to resource preemption. Therefore, it is necessary to modify the timeout interval in the **spark-defaults.conf** file of JDBCServer.

   Table 12-415 Parameter description

Parameter	Description	Default Value
spark.sql.broadcastTimeout	The timeout interval in the broadcast table of **BroadcastHashJoin**. If there are many concurrent tasks, set the parameter to a larger value or a negative number.	-1 (Numeral type. The actual value is 5 minutes.)

12.23.7.2.6 Optimizing Memory when Data Is Inserted into Dynamic Partitioned Tables

Scenario

When SparkSQL inserts data to dynamic partitioned tables, the more partitions there are, the more HDFS files a single task generates and the more memory metadata occupies. In this case, Garbage Collection (GC) is severe and Out of Memory (OOM) may occur.

Assume there are 10240 tasks and 2000 partitioned. Before the rename operation of HDFS files from a temporary directory to the target directory, there is about 29 GB FileStatus metadata.

Procedure

Insert **distribute by** followed by partition fields into dynamic partition statements.

For example:

insert into table store_returns partition (sr_returned_date_sk) select sr_return_time_sk,sr_item_sk,sr_customer_sk,sr_cdemo_sk,sr_hdemo_sk,sr_addr_sk,sr_store_sk,sr_reason_sk,sr_ticket_number,sr_return_quantity,sr_return_amt,sr_return_tax,sr_return_amt_inc_tax,sr_fee,sr_return_ship_cost,sr_refunded_cash,sr_reversed_charge,sr_store_credit,sr_net_loss,sr_returned_date_sk from ${SOURCE}.store_returns **distribute by sr_returned_date_sk**;

12.23.7.2.7 Optimizing Small Files

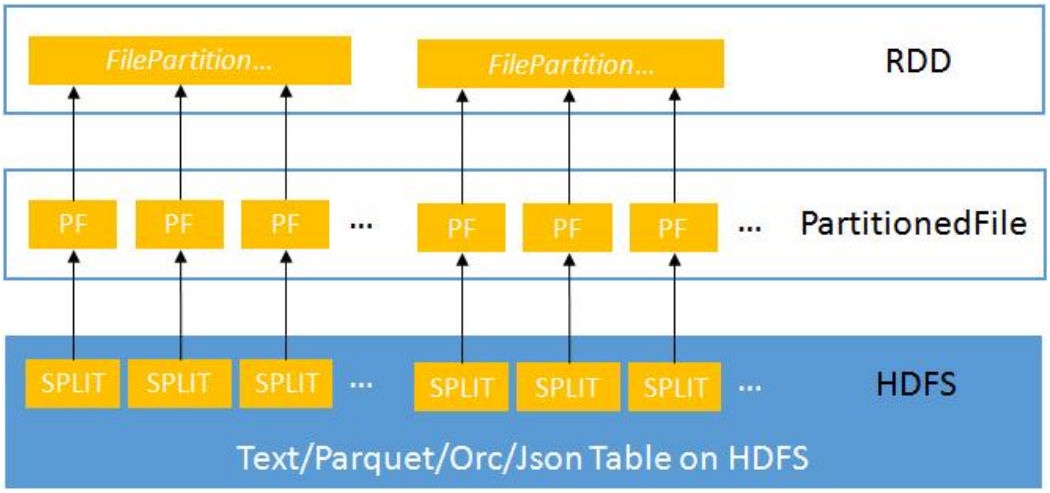
Scenario

A Spark SQL table may have many small files (far smaller than an HDFS block), each of which maps to a partition on the Spark by default. In other words, each

small file is a task. In this way, Spark has to start many such tasks. If a shuffle operation is involved in the SQL logic, the number of hash buckets soars, severely hindering system performance.

In case of massive number of small files, when DataSource creates an RDD, it splits small files in the Spark SQL table to PartitionedFiles and then merges the PartitionedFiles to a partition to avoid generating too many hash buckets during the shuffle operation. See **Figure 12-58**.

Figure 12-58 Merging small files



Procedure

If you want to enable small file optimization, configure the **spark-defaults.conf** file on the Spark client.

Table 12-416 Parameter description

Parameter	Description	Default Value
spark.sql.files.maxPartitionBytes	The maximum number of bytes that can be packed into a single partition when a file is read. Unit: byte	134217728 (128 MB)
spark.files.openCostInBytes	The estimated cost to open a file, measured by the number of bytes that can be scanned in the same time. This is used when putting multiple files into a partition. It is better to over estimate, then the partitions with small files will be faster than partitions with larger files.	4 MB

12.23.7.2.8 Optimizing the Aggregate Algorithms

Scenario

Spark SQL supports hash aggregate algorithm. Namely, use fast aggregate hashmap as cache to improve aggregate performance. The hashmap replaces the previous ColumnarBatch to avoid performance problems caused by the wide mode (multiple key or value fields) of an aggregate table.

Procedure

If you want to enable optimization of aggregate algorithm, configure following parameters in the **spark-defaults.conf** file on the Spark client.

Table 12-417 Parameter description

Parameter	Description	Default Value
spark.sql.codegen.aggregate.map.twolevel.enabled	Specifies whether to enable aggregation algorithm optimization. ● **true**: Enable ● **false**: Disable	true

12.23.7.2.9 Optimizing Datasource Tables

Scenario

Save the partition information about the datasource table to the Metastore and process partition information in the Metastore.

- Optimize the datasource tables, support syntax such as adding, deletion, and modification in the table based on partitions, improving compatibility with Hive.
- Support statements of partition tailoring and push down to the Metastore to filter unmatched partitions.

  Example:

  ```
  select count(*) from table where partCol=1;    //partCol (partition column)
  ```

  You need only to process data corresponding to partCol=1 when performing the TableScan operation in the physical plan.

Procedure

If you want to enable Datasource table optimization, configure the **spark-defaults.conf** file on the Spark client.

Table 12-418 Parameter description

Parameter	Description	Default Value
spark.sql.hive.manageFilesourcePartitions	Specifies whether to enable Metastore partition management (including datasource tables and converted Hive). ● **true** indicates enabling Metastore partition management. In this case, datasource tables are stored in Hive and Metastore is used to tailor partitions in query statements. ● **false** indicates disabling Metastore partition management.	true
spark.sql.hive.metastorePartitionPruning	Specifies whether to support pushing down predicate to Hive Metastore. ● **true** indicates supporting pushing down predicate to Hive Metastore. Only the predicate of Hive tables is supported. ● **false** indicates not supporting pushing down predicate to Hive Metastore.	true
spark.sql.hive.filesourcePartitionFileCacheSize	The cache size of the partition file metadata in the memory. All tables share a cache that can use up to specified num bytes for file metadata. This parameter is valid only when **spark.sql.hive.manageFilesourcePartitions** is set to **true**.	250 * 1024 * 1024
spark.sql.hive.convertMetastoreOrc	The processing approach of ORC tables. ● **false**: Spark SQL uses Hive SerDe to process ORC tables. ● **true**: Spark SQL uses the Spark built-in mechanism to process ORC tables.	true

12.23.7.2.10 Merging CBO

Scenario

Spark SQL supports rule-based optimization by default. However, the rule-based optimization cannot ensure that Spark selects the optimal query plan. Cost-Based Optimizer (CBO) is a technology that intelligently selects query plans for SQL statements. After CBO is enabled, the CBO optimizer performs a series of

estimations based on the table and column statistics to select the optimal query plan.

Procedure

Perform the following steps to enable CBO:

1. You need to run corresponding SQL commands to collect required table and column statistics.

   SQL commands are as follows (to be chosen as required):

   - Generate table-level statistics (table scanning):

     ANALYZE TABLE src COMPUTE STATISTICS

     This command generates **sizeInBytes** and **rowCount**.

     When you use the ANALYZE statement to collect statistics, sizes of tables not from HDFS cannot be calculated.

   - Generate table-level statistics (no table scanning):

     ANALYZE TABLE src COMPUTE STATISTICS NOSCAN

     This command generates only **sizeInBytes**. Compared with the originally generated **sizeInBytes** and **rowCount** if the **sizeInBytes** remains unchanged, **rowCount** (if any) reserves. Otherwise, **rowCount** is cleared.

   - Generate column-level statistics:

     ANALYZE TABLE src COMPUTE STATISTICS FOR COLUMNS a, b, c

     This command generates column statistics and updates table statistics for consistency. Statistics of complicated data types (such as Seq and Map) and HiveStringType cannot be generated.

   - Display statistics:

     DESC FORMATTED src

     This command displays *xxx* bytes and *xxx* rows in **Statistics** to indicate table-level statistics. You can also run the following command to display column statistics:

     DESC FORMATTED src a

   Limitation: The current statistics collection does not support statistics for partition levels for partitioned tables.

2. Configure parameters in **Table 12-419** in the **spark-defaults.conf** file on the Spark client.

   Table 12-419 Parameter description

Parameter	Description	Default Value
spark.sql.cbo.enabled	The switch to enable or disable CBO. • **true**: Enable • **false**: Disable To enable this function, ensure that statistics of related tables and columns are generated.	false

Parameter	Description	Default Value
spark.sql.cbo.joinReorder.enabled	Specifies whether to automatically adjust the sequence of consecutive inner joins by using CBO. • **true**: Enable • **false**: Disable To enable this function, ensure that statistics of related tables and columns are generated and CBO is enabled.	false
spark.sql.cbo.joinReorder.dp.threshold	Specifies the threshold of the number of tables that the sequence of consecutive inner joins is automatically adjusted by CBO. If the threshold is exceeded, the sequence of joins is not adjusted.	12

12.23.7.2.11 Optimizing SQL Query of Data of Multiple Sources

Scenario

This section describes how to enable or disable the query optimization for inter-source complex SQL.

Procedure

- (Optional) Prepare for connecting to the MPPDB data source.

  If the data source to be connected is MPPDB, a class name conflict occurs because the MPPDB Driver file **gsjdbc4.jar** and the Spark JAR package **gsjdbc4-VXXXRXXXCXXSPCXXX.jar** contain the same class name. Therefore, before connecting to the MPPDB data source, perform the following steps:

  a. Move **gsjdbc4-VXXXRXXXCXXSPCXXX.jar** from Spark. Spark running does not depend on this JAR file. Therefore, moving this JAR file to another directory (for example, the **/tmp** directory) will not affect Spark running.

     i. Log in to the Spark server and move **gsjdbc4-VXXXRXXXCXXSPCXXX.jar** from the **${BIGDATA_HOME}/FusionInsight_Spark2x_8.1.0.1/install/FusionInsight-Spark2x-3.1.1/spark/jars** directory.

     ii. Log in to the Spark client host and move **gsjdbc4-VXXXRXXXCXXSPCXXX.jar** from the **/opt/client/Spark2x/spark/jars** directory.

  b. Obtain the MPPDB Driver file **gsjdbc4.jar** from the MPPDB installation package and upload the file to the following directories:

- **/${BIGDATA_HOME}/FusionInsight_Spark2x_8.1.0.1/install/FusionInsight-Spark2x-3.1.1/spark/jars** on the Spark server.
- **/opt/client/Spark2x/spark/jars** on the Spark client.

c. Update the **/user/spark2x/jars/8.1.0.1/spark-archive-2x.zip** package stored in the HDFS.

NOTE

The version 8.1.0.1 is used as an example. Replace it with the actual version number.

i. Log in to the node where the client is installed as a client installation user. Run the following command to switch to the client installation directory, for example, **/opt/client**:

cd /opt/client

ii. Run the following command to configure environment variables:

source bigdata_env

iii. If the cluster is in security mode, run the following command to get authenticated:

kinit *Component service user*

iv. Run the following commands to create the temporary file **./tmp**, obtain **spark-archive-2x.zip** from HDFS, and decompress it to the **tmp** directory:

mkdir tmp

hdfs dfs -get /user/spark2x/jars/8.1.0.1/spark-archive-2x.zip **./**

unzip spark-archive-2x.zip -d ./tmp

v. Switch to the **tmp** directory, delete the **gsjdbc4-VXXXRXXXCXXSPCXXX.jar** file, upload the MPPDB Driver file **gsjdbc4.jar** to the **tmp** directory, and run the following command to compress the file again:

zip -r spark-archive-2x.zip *.jar

vi. Delete **spark-archive-2x.zip** from the HDFS and update the **spark-archive-2x.zip** package generated in **c.v** to the **/user/spark2x/jars/8.1.0.1/** directory in the HDFS.

hdfs dfs -rm /user/spark2x/jars/8.1.0.1/spark-archive-2x.zip

hdfs dfs -put ./spark-archive-2x.zip /user/spark2x/jars/8.1.0.1

d. Restart the Spark service. After the Spark service is restarted, restart the Spark client.

- Enable the optimization function.

  For all modules that support query pushdown, you can run the **SET** command on the **spark-beeline** client to enable the cross-source query optimization function. By default, the function is disabled.

  Pushdown configurations can be performed in dimensions of global, data sources, and tables. Commands are as follows:

  – Global (valid for all data sources):

  SET spark.sql.datasource.jdbc = project,aggregate,orderby-limit

- Data sources:

  SET spark.sql.datasource.${url} = project,aggregate,orderby-limit

- Tables:

  SET spark.sql.datasource.${url}.${table} = project,aggregate,orderby-limit

When you run the **SET** command to configure preceding parameters, you are allowed to specify multiple pushdown modules and separate them by commas. The following table lists parameters of corresponding pushdown modules.

Table 12-420 Parameters of modules

Module	Parameter Value in the SET Command
project	project
aggregate	aggregate
order by, limit over project or aggregate	orderby-limit

The following is a statement for creating an external table of MySQL:

create table if not exists pdmysql using org.apache.spark.sql.jdbc options(driver "com.mysql.jdbc.Driver", url "jdbc:mysql://ip2:3306/test", user "hive", password "*xxx*", dbtable "mysqldata");

In the preceding statement:

- ${url} = jdbc:mysql://ip2:3306/test
- ${table} = mysqldata

NOTE

- On the right of the equal sign (=) is the operators (separated by commas) to be enabled by pushdown.
- Priority: table > data source > global. If the table switch is set, the global switch of the data source is invalid for the table. If a data source switch is set, the global switch is invalid for the data source.
- The equal sign (=) is not allowed in URL. Equal signs (=) are automatically deleted in the SET clause.
- After multiple SET operations, results with different keys will not overwrite each other.

- Add functions that support query pushdown.

  In addition to query pushdown of mathematical, time, and string functions such as abs(), month(), and length(), you can run the **SET** command to add a data source that supports query pushdown. Run the following command on the Spark-beeline client:

  SET spark.sql.datasource.${datasource}.functions = fun1,fun2

- Reset the configuration set by the **SET** command.

  Currently, you can only run the **RESET** command on the **spark-beeline** client to cancel all **SET** content. After running the **RESET** command, all values in the

SET command will be cleared. Exercise caution when performing this operation.

The **SET** command is valid in the current session on the client. After the client is shut down, the content in the **SET** command turns invalid.

Alternatively, change the value of **spark.sql.locale.support** in the **spark-defaults.conf** file to **true**.

Precautions

Only MySQL, MPPDB, Hive, oracle, and PostgreSQL data sources are supported.

12.23.7.2.12 SQL Optimization for Multi-level Nesting and Hybrid Join

Scenario

This section describes the optimization suggestions for SQL statements in multi-level nesting and hybrid join scenarios.

Prerequisites

The following provides an example of complex query statements:

```
elect
s_name,
count(1) as numwait
from (
select s_name from (
select
s_name,
t2.l_orderkey,
l_suppkey,
count_suppkey,
max_suppkey
from
test2 t2 right outer join (
select
s_name,
l_orderkey,
l_suppkey from (
select
s_name,
t1.l_orderkey,
l_suppkey,
count_suppkey,
max_suppkey
from
test1 t1 join (
select
s_name,
l_orderkey,
l_suppkey
from
orders o join (
select
s_name,
l_orderkey,
l_suppkey
from
nation n join supplier s
on
s.s_nationkey = n.n_nationkey
and n.n_name = 'SAUDI ARABIA'
```

```
join lineitem l
on
s.s_suppkey = l.l_suppkey
where
l.l_receiptdate > l.l_commitdate
and l.l_orderkey is not null
) l1 on o.o_orderkey = l1.l_orderkey and o.o_orderstatus = 'F'
) l2 on l2.l_orderkey = t1.l_orderkey
) a
where
(count_suppkey > 1)
or ((count_suppkey=1)
and (l_suppkey <> max_suppkey))
) l3 on l3.l_orderkey = t2.l_orderkey
) b
where
(count_suppkey is null)
or ((count_suppkey=1)
and (l_suppkey = max_suppkey))
) c
group by
s_name
order by
numwait desc,
s_name
limit 100;
```

Procedure

Step 1 Analyze business.

Analyze business to determine whether SQL statements can be simplified through measures, for example, by combining tables to reduce the number of nesting levels and join times.

Step 2 If the SQL statements cannot be simplified, configure the driver memory.

- If SQL statements are executed through spark-submit or spark-sql, go to **Step 3**.
- If SQL statements are executed through spark-beeline, go to **Step 4**.

Step 3 During execution of SQL statements, specify the **driver-memory** parameter. An example of SQL statements is as follows:

/spark-sql --master=local[4] --driver-memory=512M -f /tpch.sql

Step 4 Before running SQL statements, change the memory size as the administrator.

1. Log in to FusionInsight Manager and choose **Cluster >** *Name of the desired cluster* **> Services** > **Spark2x** > **Configurations**.
2. On the displayed page, click **All Configurations** and search for **SPARK_DRIVER_MEMORY**.
3. Modify the **SPARK_DRIVER_MEMORY** parameter value to increase the memory size. The parameter value consists of two parts: memory size (an integer) and the unit (M or G), for example, **512M**.

----End

Reference

In the event of insufficient driver memory, the following error may be displayed during the query:

```
2018-02-11 09:13:14,683 | WARN  | Executor task launch worker for task 5 | Calling spill() on
RowBasedKeyValueBatch. Will not spill but return 0. |
org.apache.spark.sql.catalyst.expressions.RowBasedKeyValueBatch.spill(RowBasedKeyValueBatch.java:173)
2018-02-11 09:13:14,682 | WARN  | Executor task launch worker for task 3 | Calling spill() on
RowBasedKeyValueBatch. Will not spill but return 0. |
org.apache.spark.sql.catalyst.expressions.RowBasedKeyValueBatch.spill(RowBasedKeyValueBatch.java:173)
2018-02-11 09:13:14,704 | ERROR | Executor task launch worker for task 2 | Exception in task 2.0 in stage
1.0 (TID 2) | org.apache.spark.internal.Logging$class.logError(Logging.scala:91)
java.lang.OutOfMemoryError: Unable to acquire 262144 bytes of memory, got 0
        at org.apache.spark.memory.MemoryConsumer.allocateArray(MemoryConsumer.java:100)
        at org.apache.spark.unsafe.map.BytesToBytesMap.allocate(BytesToBytesMap.java:791)
        at org.apache.spark.unsafe.map.BytesToBytesMap.<init>(BytesToBytesMap.java:208)
        at org.apache.spark.unsafe.map.BytesToBytesMap.<init>(BytesToBytesMap.java:223)
        at
org.apache.spark.sql.execution.UnsafeFixedWidthAggregationMap.<init>(UnsafeFixedWidthAggregationMap.j
ava:104)
        at
org.apache.spark.sql.execution.aggregate.HashAggregateExec.createHashMap(HashAggregateExec.scala:307)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass
$GeneratedIterator.agg_doAggregateWithKeys$(Unknown Source)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIterator.processNext(Unknown
Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRowIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anonfun$8$$anon
$1.hasNext(WholeStageCodegenExec.scala:381)
        at scala.collection.Iterator$$anon$11.hasNext(Iterator.scala:408)
        at
org.apache.spark.shuffle.sort.BypassMergeSortShuffleWriter.write(BypassMergeSortShuffleWriter.java:126)
        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.scala:96)
        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.scala:53)
        at org.apache.spark.scheduler.Task.run(Task.scala:99)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:325)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:748)
```

12.23.7.3 Spark Streaming Tuning

Scenario

Streaming is a mini-batch streaming processing framework that features second-level delay and high throughput. To optimize Streaming is to improve its throughput while maintaining second-level delay so that more data can be processed per unit time.

NOTE

This section applies to the scenario where the input data source is Kafka.

Procedure

A simple streaming processing system consists of a data source, a receiver, and a processor. The data source is Kafka, the receiver is the Kafka data source receiver of Streaming, and the processor is Streaming.

Streaming optimization is to optimize the performance of the three components.

- **Data source optimization**

  In actual application scenarios, the data source stores the data in the local disks to ensure the error tolerance of the data. However, the calculation results of the Streaming are stored in the memory, and the data source may become the largest bottleneck of the streaming system.

Kafka can be optimized from the following aspects:

- Use Kafka-0.8.2 or later version that allows you to use new Producer APIs in asynchronous mode.
- Configure multiple Broker directories, multiple I/O threads, and a proper number of partitions for a topic.

For details, see section **Performance Tuning** in the Kafka open source documentation at **http://kafka.apache.org/documentation.html**.

- **Receiver optimization**

  Streaming has multiple data source receivers, such as Kafka, Flume, MQTT, and ZeroMQ. Kafka has the most receiver types and is the most mature receiver.

  Kafka provides three types of receiver APIs:

  - KafkaReceiver directly receives Kafka data. If the process is abnormal, data may be lost.
  - ReliableKafkaReceiver receives data displacement through ZooKeeper records.
  - DirectKafka reads data from each partition of Kafka through the RDD, ensuring high reliability.

  According to the implementation mechanism and test results, DirectKafka provides better performance than the other two APIs. Therefore, the DirectKafka API is recommended to implement the receiver.

  For details about the Kafka receivers and their optimization methods, see the Kafka open source documentation at **http://kafka.apache.org/documentation.html**.

- **Processor optimization**

  The bottom layer of Spark Streaming is executed by Spark. Therefore, most optimization measures for Spark can also be applied to Spark Streaming. The following is an example:

  - Data serialization
  - Memory configuration
  - Configuring DOP
  - Using the external shuffle service to improve performance

  NOTE

  Higher performance of Spark Streaming indicates lower overall reliability. Examples:

  If **spark.streaming.receiver.writeAheadLog.enable** is set to **false**, disk I/Os are reduced and performance is improved. However, because WAL is disabled, data is lost during fault recovery.

  Therefore, do not disable configuration items that ensure data reliability in production environments during Spark Streaming tuning.

- **Log archive optimization**

  The **spark.eventLog.group.size** parameter is used to group **JobHistory** logs of an application based on the specified number of jobs. Each group creates a file recording log to prevent **JobHistory** reading failures caused by an oversized log generated during the long-term running of the application. If this parameter is set to **0**, logs are not grouped.

  Most Spark Streaming jobs are small jobs and are generated at a high speed. As a result, frequent grouping is performed and a large number of small log

files are generated, consuming disk I/O resources. You are advised to increase the parameter value to, for example, **1000** or greater.

12.23.8 Common Issues About Spark2x

12.23.8.1 Spark Core

12.23.8.1.1 How Do I View Aggregated Spark Application Logs?

Question

How do I view the aggregated container logs on the page when the log aggregation function is enabled on YARN?

Answer

For details, see **Viewing Aggregated Container Logs on the Web UI**.

12.23.8.1.2 Why Is the Return Code of Driver Inconsistent with Application State Displayed on ResourceManager WebUI?

Question

Communication between ApplicationMaster and ResourceManager remains abnormal for a long time. Why is the driver return code inconsistent with application status on ResourceManager WebUI?

Answer

In yarn-client mode, Spark Driver and ApplicationMaster run as two independent processes. When Driver exits, it notifies ApplicationMaster to call the unregister API to deregister itself with ResourceManager.

This is a remote call and susceptible to network faults. If there exists a network fault, ApplicationMaster uses the retry mechanism of the Yarn client to try again. If the network is recovered before the maximum number of retries is reached, ApplicationMaster exits gracefully.

If the number and duration of retries are reached, ApplicationMaster fails to deregister itself, and ResourceManager declares ApplicationMaster to have exited forcibly and tries to restart ApplicationMaster. After the restart, if ApplicationMaster fails to connect to the exited Driver, ResourceManager flags the Application being failed.

This problem rarely occurs and it does not impact the display of application states by SparkSQL. You can also increase the number of Yarn client connections and the connection duration to reduce the probability of this event. For details about the configuration, see **http://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-common/yarn-default.xml**.

12.23.8.1.3 Why Cannot Exit the Driver Process?

Question

Why cannot exit the Driver process after running the **yarn application -kill applicationID** command to stop the Spark Streaming application?

Answer

Running the **yarn application -kill applicationID** command can only stop the SparkContext corresponding to Spark Streaming application, but cannot exit the current Driver process. If there are other permanent threads in the Driver process (for example, the spark shell is continually checking command input or Spark Streaming is continually reading data form data source), the Driver process will not be killed when the SparkContext is stopped. To exit the Driver process, you are advised to run the **kill -9 pid** command to kill the current Driver process by hand.

12.23.8.1.4 Why Does FetchFailedException Occur When the Network Connection Is Timed out

Question

On a large cluster of 380 nodes, run the ScalaSort test case in the HiBench test that runs the 29T data, and configure Executor as **--executor-cores 4**. The following abnormality is displayed:

```
org.apache.spark.shuffle.FetchFailedException: Failed to connect to /192.168.114.12:23242
    at
org.apache.spark.storage.ShuffleBlockFetcherIterator.throwFetchFailedException(ShuffleBlockFetcherIterator.s
cala:321)
    at org.apache.spark.storage.ShuffleBlockFetcherIterator.next(ShuffleBlockFetcherIterator.scala:306)
    at org.apache.spark.storage.ShuffleBlockFetcherIterator.next(ShuffleBlockFetcherIterator.scala:51)
    at scala.collection.Iterator$$anon$11.next(Iterator.scala:328)
    at scala.collection.Iterator$$anon$13.hasNext(Iterator.scala:371)
    at scala.collection.Iterator$$anon$11.hasNext(Iterator.scala:327)
    at org.apache.spark.util.CompletionIterator.hasNext(CompletionIterator.scala:32)
    at org.apache.spark.InterruptibleIterator.hasNext(InterruptibleIterator.scala:39)
    at org.apache.spark.util.collection.ExternalSorter.insertAll(ExternalSorter.scala:217)
    at org.apache.spark.shuffle.hash.HashShuffleReader.read(HashShuffleReader.scala:102)
    at org.apache.spark.rdd.ShuffledRDD.compute(ShuffledRDD.scala:90)
    at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:301)
    at org.apache.spark.rdd.RDD.iterator(RDD.scala:265)
    at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala:38)
    at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:301)
    at org.apache.spark.rdd.RDD.iterator(RDD.scala:265)
    at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala:38)
    at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:301)
    at org.apache.spark.rdd.RDD.iterator(RDD.scala:265)
    at org.apache.spark.rdd.UnionRDD.compute(UnionRDD.scala:87)
    at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:301)
    at org.apache.spark.rdd.RDD.iterator(RDD.scala:265)
    at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.scala:73)
    at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.scala:41)
    at org.apache.spark.scheduler.Task.run(Task.scala:87)
    at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:213)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
    at java.lang.Thread.run(Thread.java:745)
Caused by: java.io.IOException: Failed to connect to /192.168.114.12:23242
    at org.apache.spark.network.client.TransportClientFactory.createClient(TransportClientFactory.java:214)
    at org.apache.spark.network.client.TransportClientFactory.createClient(TransportClientFactory.java:167)
    at org.apache.spark.network.netty.NettyBlockTransferService$$anon
```

```
$1.createAndStart(NettyBlockTransferService.scala:91)
    at
org.apache.spark.network.shuffle.RetryingBlockFetcher.fetchAllOutstanding(RetryingBlockFetcher.java:140)
    at org.apache.spark.network.shuffle.RetryingBlockFetcher.access$200(RetryingBlockFetcher.java:43)
    at org.apache.spark.network.shuffle.RetryingBlockFetcher$1.run(RetryingBlockFetcher.java:170)
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511)
    at java.util.concurrent.FutureTask.run(FutureTask.java:266)
    ... 3 more
Caused by: java.net.ConnectException: Connection timed out: /192.168.114.12:23242
    at sun.nio.ch.SocketChannelImpl.checkConnect(Native Method)
    at sun.nio.ch.SocketChannelImpl.finishConnect(SocketChannelImpl.java:717)
    at io.netty.channel.socket.nio.NioSocketChannel.doFinishConnect(NioSocketChannel.java:224)
    at io.netty.channel.nio.AbstractNioChannel
$AbstractNioUnsafe.finishConnect(AbstractNioChannel.java:289)
    at io.netty.channel.nio.NioEventLoop.processSelectedKey(NioEventLoop.java:528)
    at io.netty.channel.nio.NioEventLoop.processSelectedKeysOptimized(NioEventLoop.java:468)
    at io.netty.channel.nio.NioEventLoop.processSelectedKeys(NioEventLoop.java:382)
    at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:354)
    at io.netty.util.concurrent.SingleThreadEventExecutor$2.run(SingleThreadEventExecutor.java:111)
    ... 1 more
```

Answer

When an application is run, configure the Executor parameter as **--executor-cores 4**. The degree of parallelism (DOP) is high in a single process, resulting in that the IO is highly occupied and the task works slowly.

```
16/02/26 10:04:53 INFO TaskSetManager: Finished task 2139.0 in stage 1.0 (TID 151149) in 376455 ms on
10-196-115-2 (694/153378)
```

Because running a single task takes more than 6 minutes. The network connection is timed out and the running task fails.

Set the number of cores as 1, which is **--executor-cores 1**. A task is executed smoothly in proper time (within 15s).

```
16/02/29 02:24:46 INFO TaskSetManager: Finished task 59564.0 in stage 1.0 (TID 208574) in 15088 ms on
10-196-115-6 (59515/153378)
```

Therefore, to process the task of network connection timed out and avoid such error, you can reduce the core number of a single Executor.

12.23.8.1.5 How to Configure Event Queue Size If Event Queue Overflows?

Question

How to configure the event queue size if the following Driver log information is displayed indicating that the event queue overflows?

- Common applications

  ```
  Dropping SparkListenerEvent because no remaining room in event queue.
  This likely means one of the SparkListeners is too slow and cannot keep
  up with the rate at which tasks are being started by the scheduler.
  ```

- Spark Streaming applications

  ```
  Dropping StreamingListenerEvent because no remaining room in event queue.
  This likely means one of the StreamingListeners is too slow and cannot keep
  up with the rate at which events are being started by the scheduler.
  ```

Answer

1. Stop the application. Set the configuration option **spark.event.listener.logEnable** in the Spark configuration file **spark-**

defaults.conf to **true**. And set the configuration option **spark.eventQueue.size** to **1000W**. If you need to control the logging rate (in milliseconds), also change the value of the configuration option **spark.event.listener.logRate**.

By default, the logging rate is 1000 ms, which means that one log is printed out every 1000 ms.

2. Start the application.

   The following log information is displayed, including the event consumption rate, event production rate, and **MaxSize** (maximum size of messages in the queue).

   ```
   INFO LiveListenerBus: [SparkListenerBus]:16044 events are consumed in 5000 ms.
   INFO LiveListenerBus: [SparkListenerBus]:51381 events are produced in 5000 ms, eventQueue still has 86417 events, MaxSize: 171764.
   ```

3. Change the value of the configuration option **spark.eventQueue.size** in the Spark configuration file **spark-defaults.conf** based on the **MaxSize** in the log information.

   For example, if **MaxSize** is 250000, the appropriate message queue size is 300000.

12.23.8.1.6 What Can I Do If the getApplicationReport Exception Is Recorded in Logs During Spark Application Execution and the Application Does Not Exit for a Long Time?

Question

During Spark application execution, if the driver fails to connect to ResourceManager, the following error is reported and it does not exit for a long time. What can I do?

```
16/04/23 15:31:44 INFO RetryInvocationHandler: Exception while invoking getApplicationReport of class ApplicationClientProtocolPBClientImpl over 37 after 1 fail over attempts. Trying to fail over after sleeping for 44160ms.
java.net.ConnectException: Call From vm1/192.168.39.30 to vm1:8032 failed on connection exception: java.net.ConnectException: Connection refused; For more details see:  http://wiki.apache.org/hadoop/ConnectionRefused
```

Answer

In Spark, there is a scheduled thread that listens to the status of ApplicationMaster by connecting to ResourceManager. The connection to the ResourceManager times out. As a result, the preceding error is reported and the system keeps trying to connect to the ResourceManager. In the ResourceManager, the number of retry times is limited. By default, the number of retry times is 30 and the retry interval is about 30 seconds. The preceding error is reported during each retry. The driver exits only after the number of times is exceeded.

Table 12-421 describes the retry-related configuration items in the ResourceManager.

Table 12-421 Parameter description

Parameter	Description	Default Value
yarn.resourcemanager.connect.max-wait.ms	Maximum waiting time for connecting to the ResourceManager.	900000
yarn.resourcemanager.connect.retry-interval.ms	Interval for reconnecting to the ResourceManager.	30000

Number of retries (**yarn.resourcemanager.connect.max-wait.ms/yarn.resourcemanager.connect.retry-interval.ms**) = Maximum waiting time for connecting to the ResourceManager/Interval for reconnecting to the ResourceManager

On the Spark client, modify the **conf/yarn-site.xml** file to add and configure **yarn.resourcemanager.connect.max-wait.ms** and **yarn.resourcemanager.connect.retry-interval.ms**. In this way, the number of retry times can be changed, and the Spark application can exit in advance.

12.23.8.1.7 What Can I Do If "Connection to ip:port has been quiet for xxx ms while there are outstanding requests" Is Reported When Spark Executes an Application and the Application Ends?

Question

When Spark executes an application, an error similar to the following is reported and the application ends. What can I do?

```
2016-04-20 10:42:00,557 | ERROR | [shuffle-server-2] | Connection to 10-91-8-208/10.18.0.115:57959 has been quiet for 180000 ms while there are outstanding requests. Assuming connection is dead; please adjust spark.network.timeout if this is wrong. | org.apache.spark.network.server.TransportChannelHandler.userEventTriggered(TransportChannelHandler.java:128)
2016-04-20 10:42:00,558 | ERROR | [shuffle-server-2] | Still have 1 requests outstanding when connection from 10-91-8-208/10.18.0.115:57959 is closed | org.apache.spark.network.client.TransportResponseHandler.channelUnregistered(TransportResponseHandler.java:102)
2016-04-20 10:42:00,562 | WARN  | [yarn-scheduler-ask-am-thread-pool-160] | Error sending message [message = DoShuffleClean(application_1459995017785_0108,319)] in 1 attempts | org.apache.spark.Logging$class.logWarning(Logging.scala:92)
java.io.IOException: Connection from 10-91-8-208/10.18.0.115:57959 closed
        at org.apache.spark.network.client.TransportResponseHandler.channelUnregistered(TransportResponseHandler.java:104)
        at org.apache.spark.network.server.TransportChannelHandler.channelUnregistered(TransportChannelHandler.java:94)
        at io.netty.channel.AbstractChannelHandlerContext.invokeChannelUnregistered(AbstractChannelHandlerContext.java:158)
        at io.netty.channel.AbstractChannelHandlerContext.fireChannelUnregistered(AbstractChannelHandlerContext.java:144)
        at io.netty.channel.ChannelInboundHandlerAdapter.channelUnregistered(ChannelInboundHandlerAdapter.java:53)
```

```
        at
io.netty.channel.AbstractChannelHandlerContext.invokeChannelUnregistered(AbstractChannelHandlerContex
t.java:158)
        at
io.netty.channel.AbstractChannelHandlerContext.fireChannelUnregistered(AbstractChannelHandlerContext.ja
va:144)
        at
io.netty.channel.ChannelInboundHandlerAdapter.channelUnregistered(ChannelInboundHandlerAdapter.java:5
3)
        at
io.netty.channel.AbstractChannelHandlerContext.invokeChannelUnregistered(AbstractChannelHandlerContex
t.java:158)
        at
io.netty.channel.AbstractChannelHandlerContext.fireChannelUnregistered(AbstractChannelHandlerContext.ja
va:144)
        at
io.netty.channel.ChannelInboundHandlerAdapter.channelUnregistered(ChannelInboundHandlerAdapter.java:5
3)
        at
io.netty.channel.AbstractChannelHandlerContext.invokeChannelUnregistered(AbstractChannelHandlerContex
t.java:158)
        at
io.netty.channel.AbstractChannelHandlerContext.fireChannelUnregistered(AbstractChannelHandlerContext.ja
va:144)
        at io.netty.channel.DefaultChannelPipeline.fireChannelUnregistered(DefaultChannelPipeline.java:739)
        at io.netty.channel.AbstractChannel$AbstractUnsafe$8.run(AbstractChannel.java:659)
        at io.netty.util.concurrent.SingleThreadEventExecutor.runAllTasks(SingleThreadEventExecutor.java:357)
        at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:357)
        at io.netty.util.concurrent.SingleThreadEventExecutor$2.run(SingleThreadEventExecutor.java:111)
        at java.lang.Thread.run(Thread.java:745)
2016-04-20 10:42:00,573 | INFO  | [dispatcher-event-loop-14] | Starting task 177.0 in stage 1492.0 (TID
1996351, linux-254, PROCESS_LOCAL, 2106 bytes) | org.apache.spark.Logging$class.logInfo(Logging.scala:
59)
2016-04-20 10:42:00,574 | INFO  | [task-result-getter-0] | Finished task 85.0 in stage 1492.0 (TID 1996259)
in 191336 ms on linux-254 (106/3000) | org.apache.spark.Logging$class.logInfo(Logging.scala:59)
2016-04-20 10:42:00,811 | ERROR | [Yarn application state monitor] | Yarn application has already exited
with state FINISHED! | org.apache.spark.Logging$class.logError(Logging.scala:75)
```

Answer

Symptom: The value of **spark.rpc.io.connectionTimeout** is less than the value of **spark.rpc.askTimeout**. In full GC or network delay scenarios, when the channel reaches the expiration time and still receives no response, the channel is terminated. When detecting that the channel is terminated, the AM considers the driver as disconnected, and the entire application is stopped.

Solution: Set the parameter in the **spark-defaults.conf** file on the Spark client by running the **set** command. During parameter configuration, ensure that the channel expiration time (**spark.rpc.io.connectionTimeout**) is greater than or equal to the RPC response timeout (**spark.rpc.askTimeout**).

Table 12-422 Parameter description

Parameter	Description	Default Value
spark.rpc.askTimeout	RPC response timeout. If this parameter is not set, the value of **spark.network.timeout** is used by default.	120s

12.23.8.1.8 Why Do Executors Fail to be Removed After the NodeManeger Is Shut Down?

Question

If the NodeManager is shut down with the Executor dynamic allocation enabled, the Executors on the node where the NodeManeger is shut down fail to be removed from the driver page after the idle time expires.

Answer

When the ResourceManager detects that the NodeManager is shut down, the driver has requested to kill Executors due to idle time expiry. However, the Executors cannot actually be killed because the NodeManager is shut down. The driver cannot detect the LOST events of these Executors and does not remove Executors from its Executor list. Therefore, the Executors are not removed from the driver page. This phenomenon is normal after the YARN NodeManager is shut down. The Executors will be removed after the NodeManager restarts.

12.23.8.1.9 What Can I Do If the Message "Password cannot be null if SASL is enabled" Is Displayed?

Question

ExternalShuffle is enabled for the application that runs Spark. Task loss occurs in the application because the message "java.lang.NullPointerException: Password cannot be null if SASL is enabled" is displayed. The following shows some key logs:

```
2016-05-13 12:05:27,093 | WARN  | [task-result-getter-2] | Lost task 98.0 in stage 22.1 (TID 193603, linux-173, 2): FetchFailed(BlockManagerId(13, 172.168.100.13, 27337),
org.apache.spark.shuffle.FetchFailedException: java.lang.NullPointerException: Password cannot be null if SASL is enabled
        at org.spark-project.guava.base.Preconditions.checkNotNull(Preconditions.java:208)
        at org.apache.spark.network.sasl.SparkSaslServer.encodePassword(SparkSaslServer.java:196)
        at org.apache.spark.network.sasl.SparkSaslServer$DigestCallbackHandler.handle(SparkSaslServer.java:166)
        at com.sun.security.sasl.digest.DigestMD5Server.validateClientResponse(DigestMD5Server.java:589)
        at com.sun.security.sasl.digest.DigestMD5Server.evaluateResponse(DigestMD5Server.java:244)
        at org.apache.spark.network.sasl.SparkSaslServer.response(SparkSaslServer.java:119)
        at org.apache.spark.network.sasl.SaslRpcHandler.receive(SaslRpcHandler.java:100)
        at org.apache.spark.network.server.TransportRequestHandler.processRpcRequest(TransportRequestHandler.java:128)
        at org.apache.spark.network.server.TransportRequestHandler.handle(TransportRequestHandler.java:99)
        at org.apache.spark.network.server.TransportChannelHandler.channelRead0(TransportChannelHandler.java:104)
```

Answer

The cause is that NodeManager restarts. When ExternalShuffle is used, Spark uses NodeManager to transmit shuffle data. Therefore, the memory of NodeManager may be seriously insufficient.

In the FusionInsight of the current version, the default memory of NodeManager is only 1 GB. When the data volume of Spark tasks is large (greater than 1 TB), the memory is severely insufficient and the message response is slow. As a result, the FusionInsight health check determines that the NodeManager process exits and forcibly restarts the NodeManager, causing the preceding problem.

Solution

Adjust the memory of the NodeManager. If the data volume is large (greater than 1 TB), the memory of NodeManager must be greater than 4 GB.

12.23.8.1.10 What Should I Do If the Message "Failed to CREATE_FILE" Is Displayed in the Restarted Tasks When Data Is Inserted Into the Dynamic Partition Table?

Question

When inserting data into the dynamic partition table, a large number of shuffle files are damaged due to the disk disconnection, node error, and the like. In this case, why the message **Failed to CREATE_FILE** is displayed in the restarted tasks?

```
2016-06-25 15:11:31,323 | ERROR | [Executor task launch worker-0] | Exception in task 15.0 in stage 10.1 (TID 1258) | org.apache.spark.Logging$class.logError(Logging.scala:96)
org.apache.hadoop.hive.ql.metadata.HiveException:
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.hdfs.protocol.AlreadyBeingCreatedException):
Failed to CREATE_FILE /user/hive/warehouse/testdb.db/we
b_sales/.hive-staging_hive_2016-06-25_15-09-16_999_8137121701603617850-1/-ext-10000/_temporary/0/
_temporary/attempt_201606251509_0010_m_000015_0/ws_sold_date=1999-12-17/part-00015 for
DFSClient_attempt_2016
06251509_0010_m_000015_0_353134803_151 on 10.1.1.5 because this file lease is currently owned by
DFSClient_attempt_201606251509_0010_m_000015_0_-848353830_156 on 10.1.1.6
```

Answer

The last step of inserting data into the dynamic partition table is to read shuffle files and then write the data to the mapped partition files.

After a large number of shuffle files are damaged, a large number of tasks fail, causing the restart of jobs. Before the restart of jobs, Spark closes the handles that write table partition files. However, the HDFS cannot process the scenario of batch tasks closing handles. After tasks restart next time, the handles are not released in a timely manner on the NameNode. As a result, the message **Failed to CREATE_FILE** is displayed.

This error only occurs when a large number of shuffle files are damaged. The tasks will restart after the error occurs and the restart can be completed within milliseconds.

12.23.8.1.11 Why Tasks Fail When Hash Shuffle Is Used?

Question

When Hash shuffle is used to run a job that consists of 1000000 map tasks x 100000 reduce tasks, run logs report many message failures and Executor heartbeat timeout, leading to task failures. Why does this happen?

Answer

During the shuffle process, Hash shuffle just writes the data of different reduce partitions to their respective disk files according to hash results without sorting the data.

If there are many reduce partitions, a large number of disk files will be generated. In your case, 10^11 shuffle files, that is, 1000000 * 100000 shuffle files, will be generated. The sheer number of disk files will have a great impact on the file read and write performance. In addition, the operations such as sorting and compressing will consume a large amount of temporary memory space because a large number of file handles are open, presenting great challenges to memory

management and garbage collection and incurring the possibility that the Executor fails to respond to Driver.

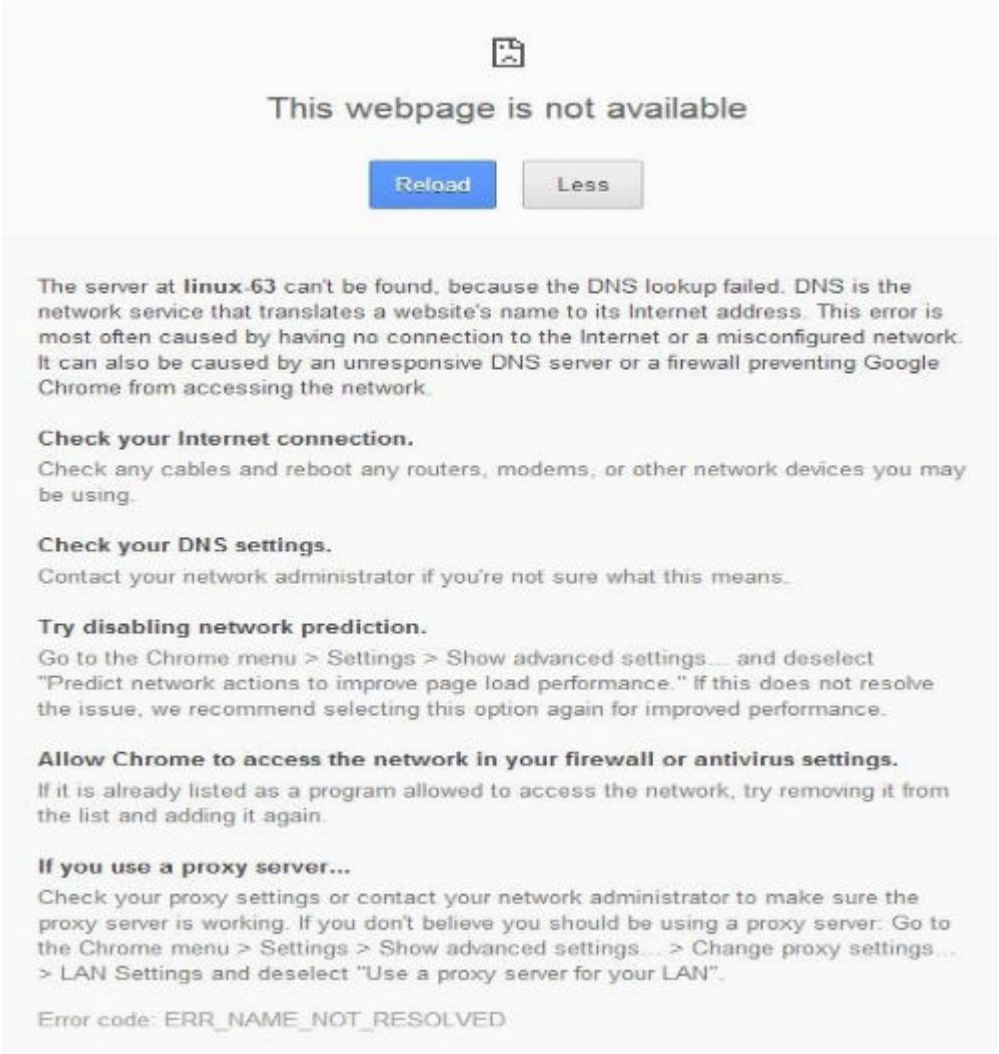
Sort shuffle, instead of Hash shuffle, is recommended to run a job.

12.23.8.1.12 What Can I Do If the Error Message "DNS query failed" Is Displayed When I Access the Aggregated Logs Page of Spark Applications?

Question

When the **http(s)://<spark ip>:<spark port>** mode is used to access the Spark JobHistory page, if the displayed Spark JobHistory page is not the page of FusionInsight Manager (the URL of FusionInsight Manager is similar to **https://<oms ip>:20026/Spark2x/JobHistory2x/xx/**), click an application and click **AggregatedLogs**, click the logs of an executor to be viewed. An error message in **Figure 12-59** is displayed.

Figure 12-59 DNS query failure



Answer

Cause: The domain name is not added to the **hosts** file of the Windows OS in the pop-up URL (for example, **https://<hostname>:20026/Spark2x/JobHistory2x/xx/history/application_xxx/jobs/**). As a result, the DNS query fails and the web page cannot be displayed.

Solution:

- You are advised to visit **Spark JobHistory** page using the FusionInsight Manager..
- If you do not want to access the **Spark JobHistory** page using the FusionInsight Manager, change **<hostname>** in the URL to the IP address or add the domain name to the **hosts** file of the Windows OS.

12.23.8.1.13 What Can I Do If Shuffle Fetch Fails Due to the "Timeout Waiting for Task" Exception?

Question

When I execute a 100 TB TPC-DS test suite in the JDBCServer mode, the "Timeout waiting for task" is displayed. As a result, shuffle fetch fails, the stage keeps retrying, and the task cannot be completed properly. What can I do?

Answer

The ShuffleService function is used in JDBCServer mode. In the reduce phase, all executors obtain data from NodeManager. When the data volume reaches a level (more than 10 TB), the NodeManager may reach the bottleneck (ShuffleService is in the NodeManager process). As a result, some tasks for obtaining data time out. Therefore, the problem occurs.

You are advised to disable ShuffleService for Spark tasks whose data volume is greater than 10 TB. That is, set **spark.shuffle.service.enabled** in the **Spark-defaults.conf** configuration file to **false**.

12.23.8.1.14 Why Does the Stage Retry due to the Crash of the Executor?

Question

When I run Spark tasks with a large data volume, for example, 100 TB TPCDS test suite, why does the Stage retry due to Executor loss sometimes? The message "Executor 532 is lost rpc with driver, but is still alive, going to kill it" is displayed, indicating that the loss of the Executor is caused by a JVM crash.

The log of the key JVM crash is as follows:

```
#
# A fatal error has been detected by the Java Runtime Environment:
#
#  Internal Error (sharedRuntime.cpp:834), pid=241075, tid=140476258551552
#  fatal error: exception happened outside interpreter, nmethods and vtable stubs at pc
0x00007fcda9eb8eb1
```

Answer

This error does not affect services. This error is caused by defects of the Oracle JVM, but not the platform code. There is the fault tolerance mechanism for Executors in Spark: the Stage retries in case of an Executor crash to ensure the success execution of tasks.

12.23.8.1.15 Why Do the Executors Fail to Register Shuffle Services During the Shuffle of a Large Amount of Data?

Question

When more than 50 terabytes of data is shuffled, some executors fail to register shuffle services due to timeout. The shuffle tasks then fail. Why? The error log is as follows:

```
2016-10-19 01:33:34,030 | WARN | ContainersLauncher #14 | Exception from container-launch with
container ID: container_e1452_1476801295027_2003_01_004512 and exit code: 1 |
LinuxContainerExecutor.java:397
ExitCodeException exitCode=1:
at org.apache.hadoop.util.Shell.runCommand(Shell.java:561)
at org.apache.hadoop.util.Shell.run(Shell.java:472)
at org.apache.hadoop.util.Shell$ShellCommandExecutor.execute(Shell.java:738)
at
org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor.launchContainer(LinuxContainerExecuto
r.java:381)
at
org.apache.hadoop.yarn.server.nodemanager.containermanager.launcher.ContainerLaunch.call(ContainerLaun
ch.java:312)
at
org.apache.hadoop.yarn.server.nodemanager.containermanager.launcher.ContainerLaunch.call(ContainerLaun
ch.java:88)
at java.util.concurrent.FutureTask.run(FutureTask.java:266)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
at java.lang.Thread.run(Thread.java:745)
2016-10-19 01:33:34,031 | INFO | ContainersLauncher #14 | Exception from container-launch. |
ContainerExecutor.java:300
2016-10-19 01:33:34,031 | INFO | ContainersLauncher #14 | Container id:
container_e1452_1476801295027_2003_01_004512 | ContainerExecutor.java:300
2016-10-19 01:33:34,031 | INFO | ContainersLauncher #14 | Exit code: 1 | ContainerExecutor.java:300
2016-10-19 01:33:34,031 | INFO | ContainersLauncher #14 | Stack trace: ExitCodeException exitCode=1: |
ContainerExecutor.java:300
```

Answer

The imported data exceeds 50 TB, which exceeds the shuffle processing capability. The shuffle may fail to respond to the registration request of an executor in a timely manner due to the heavy load.

The timeout interval for an executor to register the shuffle service is 5 seconds. The maximum number of retries is 3. This parameter is not configurable.

You are advised to increase the number of task retry times and the number of allowed executor failure times.

Configure the following parameters in the **spark-defaults.conf** file on the client: If **spark.yarn.max.executor.failures** does not exist, manually add it.

Table 12-423 Parameter Description

Parameter	Description	Default Value
spark.task.maxFailures	Specifies task retry times.	4
spark.yarn.max.executor.failures	Specifies executor failure attempt times. Set **spark.dynamicAllocation.enabled** to **false**, to disable the dynamic allocation of executors.	numExecutors * 2, with minimum of 3
	Specifies executor failure attempt times. Set **spark.dynamicAllocation.enabled** to **true**, to enable the dynamic allocation of executors.	3

12.23.8.1.16 Why Does the Out of Memory Error Occur in NodeManager During the Execution of Spark Applications

Question

During the execution of Spark applications, if the YARN External Shuffle service is enabled and there are too many shuffle tasks, the **java.lang.OutofMemoryError: Direct buffer Memory** error occurs, indicating insufficient memory. The error log is as follows:

```
2016-12-06 02:01:00,768 | WARN  | shuffle-server-38 | Exception in connection from /192.168.101.95:53680 | TransportChannelHandler.java:79
io.netty.handler.codec.DecoderException: java.lang.OutOfMemoryError: Direct buffer memory
        at io.netty.handler.codec.ByteToMessageDecoder.channelRead(ByteToMessageDecoder.java:153)
        at io.netty.channel.AbstractChannelHandlerContext.invokeChannelRead(AbstractChannelHandlerContext.java:333)
        at io.netty.channel.AbstractChannelHandlerContext.fireChannelRead(AbstractChannelHandlerContext.java:319)
        at io.netty.channel.DefaultChannelPipeline.fireChannelRead(DefaultChannelPipeline.java:787)
        at io.netty.channel.nio.AbstractNioByteChannel$NioByteUnsafe.read(AbstractNioByteChannel.java:130)
        at io.netty.channel.nio.NioEventLoop.processSelectedKey(NioEventLoop.java:511)
        at io.netty.channel.nio.NioEventLoop.processSelectedKeysOptimized(NioEventLoop.java:468)
        at io.netty.channel.nio.NioEventLoop.processSelectedKeys(NioEventLoop.java:382)
        at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:354)
        at io.netty.util.concurrent.SingleThreadEventExecutor$2.run(SingleThreadEventExecutor.java:116)
        at java.lang.Thread.run(Thread.java:745)
Caused by: java.lang.OutOfMemoryError: Direct buffer memory
        at java.nio.Bits.reserveMemory(Bits.java:693)
        at java.nio.DirectByteBuffer.<init>(DirectByteBuffer.java:123)
        at java.nio.ByteBuffer.allocateDirect(ByteBuffer.java:311)
        at io.netty.buffer.PoolArena$DirectArena.newChunk(PoolArena.java:434)
        at io.netty.buffer.PoolArena.allocateNormal(PoolArena.java:179)
        at io.netty.buffer.PoolArena.allocate(PoolArena.java:168)
        at io.netty.buffer.PoolArena.reallocate(PoolArena.java:277)
```

```
    at io.netty.buffer.PooledByteBuf.capacity(PooledByteBuf.java:108)
    at io.netty.buffer.AbstractByteBuf.ensureWritable(AbstractByteBuf.java:251)
    at io.netty.buffer.AbstractByteBuf.writeBytes(AbstractByteBuf.java:849)
    at io.netty.buffer.AbstractByteBuf.writeBytes(AbstractByteBuf.java:841)
    at io.netty.buffer.AbstractByteBuf.writeBytes(AbstractByteBuf.java:831)
    at io.netty.handler.codec.ByteToMessageDecoder.channelRead(ByteToMessageDecoder.java:146)
    ... 10 more
```

Answer

In the Shuffle Service of YARN, the number of started threads are twice of the number of available CPU cores. The default size of direct buffer memory is 128 MB. If there are too many shuffle tasks connected at the same time, the direct buffer memory allocated to each thread service is insufficient. For example, if there are 40 CPU cores and there are 80 threads started by the Shuffle Service of YARN, the direct buffer memory allocated to each thread is less than 2 MB.

To solve this problem, increase the directory buffer memory based on the number of CPU cores in NodeManager. For example, if there are 40 of CPU cores, increase the direct buffer memory to 512 MB, that is, configure the **GC_OPTS** parameter of NodeManager as follows:

-XX:MaxDirectMemorySize=512M

NOTE

By default, **-XX:MaxDirectMemorySize** is not configured in the **GC_OPTS** parameter. To configure it, you need to add it to the **GC_OPTS** parameter as an custom option.

To configure the **GC_OPTS** parameter, log in to FusionInsight Manager, choose **Cluster >** *Name of the desired cluster* **> Services** > **Yarn** > **Configurations**, click **All Configurations**, and choose **NodeManager > System**, and then modify the **GC_OPTS** parameter.

Table 12-424 Parameter description

Parameter	Description	Default Value
GC_OPTS	The GC parameter of YARN NodeManger.	128M

12.23.8.1.17 Why Does the Realm Information Fail to Be Obtained When SparkBench is Run on HiBench for the Cluster in Security Mode?

Question

Execution of the sparkbench task (for example, Wordcount) of HiBench6 fails. The bench.log indicates that the Yarn task fails to be executed. The failure information displayed on the Yarn UI is as follows:

```
Exception in thread "main" org.apache.spark.SparkException: Unable to load YARN support
  at org.apache.spark.deploy.SparkHadoopUtil$.liftedTree1$1(SparkHadoopUtil.scala:390)
  at org.apache.spark.deploy.SparkHadoopUtil$.yarn$lzycompute(SparkHadoopUtil.scala:385)
  at org.apache.spark.deploy.SparkHadoopUtil$.yarn(SparkHadoopUtil.scala:385)
  at org.apache.spark.deploy.SparkHadoopUtil$.get(SparkHadoopUtil.scala:410)
  at org.apache.spark.deploy.yarn.ApplicationMaster$.main(ApplicationMaster.scala:796)
  at org.apache.spark.deploy.yarn.ExecutorLauncher$.main(ApplicationMaster.scala:821)
  at org.apache.spark.deploy.yarn.ExecutorLauncher.main(ApplicationMaster.scala)
```

```
Caused by: java.lang.IllegalArgumentException: Can't get Kerberos realm
 at org.apache.hadoop.security.HadoopKerberosName.setConfiguration(HadoopKerberosName.java:65)
 at org.apache.hadoop.security.UserGroupInformation.initialize(UserGroupInformation.java:288)
 at org.apache.hadoop.security.UserGroupInformation.setConfiguration(UserGroupInformation.java:336)
 at org.apache.spark.deploy.SparkHadoopUtil.<init>(SparkHadoopUtil.scala:51)
 at org.apache.spark.deploy.yarn.YarnSparkHadoopUtil.<init>(YarnSparkHadoopUtil.scala:49)
 at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
 at sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
 at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
 at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
 at java.lang.Class.newInstance(Class.java:442)
 at org.apache.spark.deploy.SparkHadoopUtil$.liftedTree1$1(SparkHadoopUtil.scala:387)
 ... 6 more
Caused by: java.lang.reflect.InvocationTargetException
 at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
 at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
 at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
 at java.lang.reflect.Method.invoke(Method.java:498)
 at org.apache.hadoop.security.authentication.util.KerberosUtil.getDefaultRealm(KerberosUtil.java:88)
 at org.apache.hadoop.security.HadoopKerberosName.setConfiguration(HadoopKerberosName.java:63)
 ... 16 more
Caused by: KrbException: Cannot locate default realm
 at sun.security.krb5.Config.getDefaultRealm(Config.java:1029)
 ... 22 more
```

Answer

In C80SPC200 and later, the file stored in the **/etc/krb5.conf** directory is no longer replaced during cluster installation. Instead, the file is stored in the corresponding path on the client through parameter configurations, and HiBench does not reference the client configuration file. Solution: Use the file stored in the **/opt/client/KrbClient/kerberos/var/krb5kdc/krb5.conf** directory on the client to overwrite that in the **/etc/krb5.conf** directories of all nodes. Make a backup before the overwriting.

12.23.8.2 Spark SQL and DataFrame

12.23.8.2.1 What Do I have to Note When Using Spark SQL ROLLUP and CUBE?

Question

Suppose that there is a table src(d1, d2, m) with the following data:

```
1 a 1
1 b 1
2 b 2
```

The results for statement "select d1, sum(d1) from src group by d1, d2 with rollup" are shown as below:

```
NULL 0
1    2
2    2
1    1
1    1
2    2
```

Why the first line of the above results is (NULL,0), rather than (NULL,4)?

Answer

When conducting the rollup and cube operation, we usually perform the dimension-based analysis and what we need is the measurement result, so we would not conduct aggregation operation on the dimension.

Suppose that there is a table src(d1, d2, m), so the statement 1 "select d1, sum(m) from src group by d1, d2 with rollup" conducts the rollup operation on the dimension d1 and d2 to compute the result of m. It has actual business meaning, and its results are in line with the expectation. However, the statement 2 "select d1, sum(d1) from src group by d1, d2 with rollup" cannot be explained from the business perspective. For the statement 2, the result for all aggregations (sum/avg/max/min) is 0.

NOTE

Only when there is an aggregation operation for fields in "group by" in the rollup and cube operation, the result is 0. For non-rollup and non-cube operations, the result will be in line with the expectation.

12.23.8.2.2 Why Spark SQL Is Displayed as a Temporary Table in Different Databases?

Question

Why temporary tables of the previous database are displayed after the database is switched?

1. Create a temporary DataSource table, for example:
```
create temporary table ds_parquet
using org.apache.spark.sql.parquet
options(path '/tmp/users.parquet');
```
2. Switch to another database, and run ***show tables***. The temporary table created in the previous table is displayed.
```
0: jdbc:hive2://192.168.169.84:22550/default> show tables;
+-----------------+--------------+--+
|    tableName    | isTemporary  |
+-----------------+--------------+--+
| ds_parquet      | true         |
| cmb_tbl_carbon  | false        |
+-----------------+--------------+--+
2 rows selected (0.109 seconds)
0: jdbc:hive2://192.168.169.84:22550/default>
```

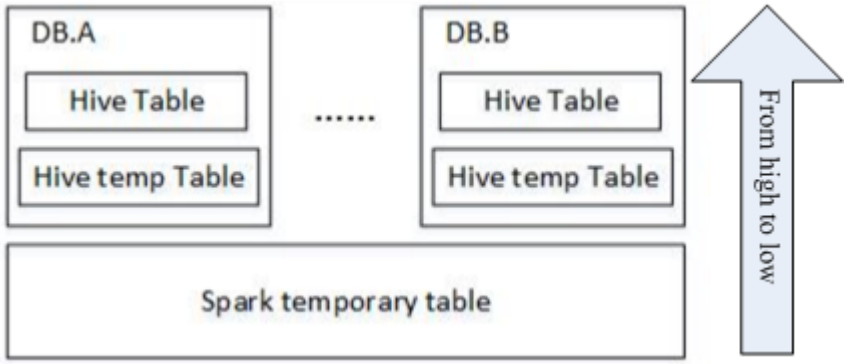
Answer

The table management hierarchy of Spark is shown in **Figure 12-60**. The lowest layer stores all temporary DataSource tables. There is no such concept as database at this layer. DataSource tables are visible in various databases.

The MetaStore of Hive is located at the upper layer. This layer distinguishes among databases. In each database, there are two types of Hive table, permanent and temporary. Therefore, Spark supports data tables of the same name at three layers.

During query, SparkSQL first checks for temporary Spark tables, then temporary Hive tables in the current database, and at last the permanent tables in the current database.

Figure 12-60 Spark table management hierarchy



When a session quits, temporary tables related to the user operation are automatically deleted. Manual deletion of temporary files is not recommended.

When deleting temporary files, use the same priority as that for query. The priorities are temporary Spark table, temporary Hive table, and permanent Hive table ranging from high to low. If you want to directly delete Hive tables but not temporary Spark tables, you can directly use the ***drop table DbName.TableName*** command.

12.23.8.2.3 How to Assign a Parameter Value in a Spark Command?

Question

Is it possible to assign parameter values through Spark commands, in addition to through a user interface or a configuration file?

Answer

Spark configuration options can be defined either in a configuration file or in Spark commands.

To assign a parameter value, run the --conf command on a Spark client. The parameter value takes effect immediately after the command is run.

The command format is --conf + parameter name + parameter value. Example command:

--conf spark.eventQueue.size=50000

12.23.8.2.4 What Directory Permissions Do I Need to Create a Table Using SparkSQL?

Question

The following error information is displayed when a new user creates a table using SparkSQL:

```
0: jdbc:hive2://192.168.169.84:22550/default> create table testACL(c string);
Error: org.apache.spark.sql.execution.QueryExecutionException: FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.DDLTask. MetaException(message:Got exception:
org.apache.hadoop.security.AccessControlException
Permission denied: user=testACL, access=EXECUTE, inode="/user/hive/warehouse/
```

```
testacl":spark:hadoop:drwxrwx---
    at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkAccessAcl(FSPermissionChecker.java:403
)
    at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.check(FSPermissionChecker.java:306)
    at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkTraverse(FSPermissionChecker.java:259)
    at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:20
5)
    at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:19
0)
    at org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPermission(FSDirectory.java:1710)
    at
org.apache.hadoop.hdfs.server.namenode.FSDirStatAndListingOp.getFileInfo(FSDirStatAndListingOp.java:109)
    at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.getFileInfo(FSNamesystem.java:3762)
    at
org.apache.hadoop.hdfs.server.namenode.NameNodeRpcServer.getFileInfo(NameNodeRpcServer.java:1014)
    at
org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolServerSideTranslatorPB.getFileInfo(ClientNamen
odeProtocolServerSideTranslatorPB.java:853)
    at org.apache.hadoop.hdfs.protocol.proto.ClientNamenodeProtocolProtos$ClientNamenodeProtocol
$2.callBlockingMethod(ClientNamenodeProtocolProtos.java)
    at org.apache.hadoop.ipc.ProtobufRpcEngine$Server$ProtoBufRpcInvoker.call(ProtobufRpcEngine.java:616)
    at org.apache.hadoop.ipc.RPC$Server.call(RPC.java:973)
    at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2089)
    at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2085)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.Subject.doAs(Subject.java:422)
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1675)
    at org.apache.hadoop.ipc.Server$Handler.run(Server.java:2083)
) (state=,code=0)
```

Answer

When you create a table using Spark SQL, the interface of Hive is called by the underlying system and a directory named after the table will be created in the **/user/hive/warehouse** directory. Therefore, you must have the permissions to read, write, and execute the **/user/hive/warehouse** directory or the group permission of Hive.

The**/user/hive/warehouse** is specified by the hive.metastore.warehouse.dir parameter.

12.23.8.2.5 Why Do I Fail to Delete the UDF Using Another Service?

Question

Why do I fail to delete the UDF using another service, for example, delete the UDF created by Hive using Spark SQL.

Answer

The UDF can be created using any of the following services:

1. Hive client.
2. JDBCServer API. You can connect JDBCServer to Spark Beeline or JDBC client code, and run SQL statements to create the UDF.
3. spark-sql.

The scenarios in which the UDF failed to be deleted may be as follows:

- If you use Spark Beeline to delete the UDF created by other services, you must restart the JDBCServer before the deletion. Otherwise, the deletion fails. If you use spark-sql to delete the UDF created by other services, you must restart the spark-sql before the deletion. Otherwise, the deletion fails.

  Cause: After the UDF is created, if the JDBCServer or the spark-sql has not been restarted, the newly created UDF will not be saved by the FunctionRegistry object in the thread where Spark locates. As a result, the UDF failed to be deleted.

  Solution: Restart the JDBCServer and spark-sql of the Spark client and delete the UDF.

- When creating UDF on the Hive client, the **add jar** command (e.g. **add jar /opt/test/two_udfs.jar**) is used to add the **.jar** package instead of specifying the path of **.jar** package in creating UDF statement. As a result, the **ClassNotfound** error occurs when you use other services to delete the UDF.

  Cause: When you use a service to delete the UDF, the service will load the class that corresponds to the UDF to obtain the UDF. However, the **.jar** package is added by the ***add jar*** command and jar package does not exist in the classpath of other services. As a result, the **ClassNotfound** error occurs and the UDF failed to be deleted.

  Solution: The UDF created using the preceding approach must be deleted using the same approach. No other approaches are allowed.

12.23.8.2.6 Why Cannot I Query Newly Inserted Data in a Parquet Hive Table Using SparkSQL?

Question

Why cannot I query newly inserted data in a parquet Hive table using SparkSQL? This problem occurs in the following scenarios:

1. For partitioned tables and non-partitioned tables, after data is inserted on the Hive client, the latest inserted data cannot be queried using SparkSQL.
2. After data is inserted into a partitioned table using SparkSQL, if the partition information remains unchanged, the newly inserted data cannot be queried using SparkSQL.

Answer

To improve Spark performance, parquet metadata is cached. When the parquet table is updated by Hive or another means, the cached metadata remains unchanged, resulting in SparkSQL failing to query the newly inserted data.

For a parquet Hive partition table, if the partition information remains unchanged after data is inserted, the cached metadata is not updated. As a result, the newly inserted data cannot be queried by SparkSQL.

To solve the query problem, update metadata before starting a Spark SQL query.

REFRESH TABLE table_name;

table_name indicates the name of the table to be updated. The table must exist. Otherwise, an error is reported.

When the query statement is executed, the latest inserted data can be obtained.

For details, visit **https://archive.apache.org/dist/spark/docs/3.1.1/sql-programming-guide.html#metadata-refreshing**.

12.23.8.2.7 How to Use Cache Table?

Question

What is cache table used for? Which point should I pay attention to while using cache table?

Answer

Spark SQL caches tables into memory so that data can be directly read from memory instead of disks, reducing memory overhead due to disk reads.

Note that cached tables consume Executor's memory. This means that caching large or many tables compromises Executor's stability even if compressed storage has been used to reduce memory overhead as much as possible.

If it is no longer necessary to accelerate data query by means of cache table, run the following command to uncache tables to free up memory:

uncache table *table_name*

NOTE

The Storage tab page of the Spark Driver user interface displays the cached tables.

12.23.8.2.8 Why Are Some Partitions Empty During Repartition?

Question

During the repartition operation, the number of blocks (**spark.sql.shuffle.partitions**) is set to 4,500, and the number of keys used by repartition exceeds 4,000. It is expected that data corresponding to different keys can be allocated to different partitions. However, only 2,000 partitions have data, and data corresponding to different keys is allocated to the same partition.

Answer

This is normal.

The partition to which data is distributed is obtained by performing a modulo operation on hashcode of a key. Different hashcodes may have the same modulo result. In this case, data is distributed to the same partition, as a result, some partitions do not have data, and some partitions have data corresponding to multiple keys.

You can adjust the value of **spark.sql.shuffle.partitions** to adjust the cardinality during modulo operation and improve the unevenness of data blocks. After multiple verifications, it is found that the effect is good when the parameter is set to a prime number or an odd number.

Configure the following parameters in the **spark-defaults.conf** file on the Driver client.

Table 12-425 Parameter Description

Parameter	Description	Default Value
spark.sql.shuffle.partitions	Number of shuffle data blocks during the shuffle operation.	200

12.23.8.2.9 Why Does 16 Terabytes of Text Data Fails to Be Converted into 4 Terabytes of Parquet Data?

Question

When the default configuration is used, 16 terabytes of text data fails to be converted into 4 terabytes of parquet data, and the error information below is displayed. Why?

```
Job aborted due to stage failure: Task 2866 in stage 11.0 failed 4 times, most recent failure: Lost task
2866.6 in stage 11.0 (TID 54863, linux-161, 2): java.io.IOException: Failed to connect to /10.16.1.11:23124
at org.apache.spark.network.client.TransportClientFactory.createClient(TransportClientFactory.java:214)
at org.apache.spark.network.client.TransportClientFactory.createClient(TransportClientFactory.java:167)
at org.apache.spark.network.netty.NettyBlockTransferService$$anon
$1.createAndStart(NettyBlockTransferService.scala:92)
```

Table 12-426 lists the default configuration.

Table 12-426 Parameter Description

Parameter	Description	Default Value
spark.sql.shuffle.partitions	Number of shuffle data blocks during the shuffle operation.	200
spark.shuffle.sasl.timeout	Timeout interval of SASL authentication for the shuffle operation. Unit: second	120s
spark.shuffle.io.connectionTimeout	Timeout interval for connecting to a remote node during the shuffle operation. Unit: second	120s
spark.network.timeout	Timeout interval for all network connection operations. Unit: second	360s

Answer

The current data volume is 16 TB, but the number of partitions is only 200. As a result, each task is overloaded and the preceding problem occurs.

To solve the preceding problem, you need to adjust the parameters.

- Increase the number of partitions to divide the task into smaller ones.

- Increase the timeout interval during task execution.

Configure the following parameters in the **spark-defaults.conf** file on the client:

Table 12-427 Parameter Description

Parameter	Description	Recommended Value
spark.sql.shuffle.partitions	Number of shuffle data blocks during the shuffle operation.	4501
spark.shuffle.sasl.timeout	Timeout interval of SASL authentication for the shuffle operation. Unit: second	2000s
spark.shuffle.io.connectionTimeout	Timeout interval for connecting to a remote node during the shuffle operation. Unit: second	3000s
spark.network.timeout	Timeout interval for all network connection operations. Unit: second	360s

12.23.8.2.10 Why the Operation Fails When the Table Name Is TABLE?

Question

When the table name is set to **table**, why the error information similar to the following is displayed after the **drop table table** command or other command is run?

```
16/07/12 18:56:29 ERROR SparkSQLDriver: Failed in [drop table table]
java.lang.RuntimeException: [1.1] failure: identifier expected
table
^
at scala.sys.package$.error(package.scala:27)
at org.apache.spark.sql.catalyst.SqlParserTrait$class.parseTableIdentifier(SqlParser.scala:56)
at org.apache.spark.sql.catalyst.SqlParser$.parseTableIdentifier(SqlParser.scala:485)
```

Answer

The word table is a keyword of Spark SQL statements and must not be used as a table name.

12.23.8.2.11 Why Is a Task Suspended When the ANALYZE TABLE Statement Is Executed and Resources Are Insufficient?

Question

When the ***analyze table*** statement is executed using spark-sql, the task is suspended and the information below is displayed. Why?

```
spark-sql> analyze table hivetable2 compute statistics;
Query ID = root_20160716174218_90f55869-000a-40b4-a908-533f63866fed
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
16/07/20 17:40:56 WARN JobResourceUploader: Hadoop command-line option parsing not performed.
Implement the Tool interface and execute your application with ToolRunner to remedy this.
Starting Job = job_1468982600676_0002, Tracking URL = http://10-120-175-107:8088/proxy/
application_1468982600676_0002/
Kill Command = /opt/hadoopclient/HDFS/hadoop/bin/hadoop job  -kill job_1468982600676_0002
```

Answer

When the statement is executed, the SQL statement starts the ***analyze table hivetable2 compute statistics*** MapReduce tasks. On the ResourceManager Web UI of Yarn, the task is not executed due to insufficient resources. As a result, the task is suspended.

Figure 12-61 ResourceManager Web UI

You are advised to add **noscan** when running the ***analyze table*** statement. The function of this statement is the same as that of the ***analyze table hivetable2 compute statistics*** statement. The command is as follows:

```
spark-sql> analyze table hivetable2 compute statistics noscan
```

This command does not start MapReduce tasks and does not occupy Yarn resources. Therefore, the tasks can be executed.

12.23.8.2.12 If I Access a parquet Table on Which I Do not Have Permission, Why a Job Is Run Before "Missing Privileges" Is Displayed?

Question

If I access a parquet table on which I do not have permission, why a job is run before "Missing Privileges" is displayed?

Answer

The execution sequence of Spark SQL statement parse the table in the statement first, then obtain the metadata in the table, and finally check the permission.

The metadata of a parquet table contains the Split information (which is read by HDFS API) about files. If the table contains many files, the HDFS API reads data in serial mode, in which degrades the performance. If the number of files in the table exceeds the threshold *spark.sql.sources.parallelSplitDiscovery.threshold*, a job will be generated to use Executor to read the data in parallel mode.

The permission authentication is executed after the metadata is obtained. Therefore, when the number of files in the table exceeds the threshold, a job is run before the permission authentication error message **Missing Privileges**.

12.23.8.2.13 Why Do I Fail to Modify MetaData by Running the Hive Command?

Question

When do I fail to modify the metadata in the datasource and Spark on HBase table by running the Hive command?

Answer

The current Spark version does not support modifying the metadata in the datasource and Spark on HBase tables by running the Hive command.

12.23.8.2.14 Why Is "RejectedExecutionException" Displayed When I Exit Spark SQL?

Question

After successfully running Spark tasks with large data volume, for example, 2-TB TPCDS test suite, why is the abnormal stack information **"RejectedExecutionException"** displayed sometimes? The log is as follows:

```
16/07/16 10:19:56 ERROR TransportResponseHandler: Still have 2 requests outstanding when connection from linux-192/10.1.1.5:59250 is closed
java.util.concurrent.RejectedExecutionException: Task scala.concurrent.impl.CallbackRunnable@5fc1ab rejected from java.util.concurrent.ThreadPoolExecutor@52fa7e19[Terminated, pool size = 0, active threads = 0, queued tasks = 0, completed tasks = 3025]
```

Answer

When Spark SQL is closed, the application and the message channel are closed. If there are unprocessed messages, the connection should be closed to rectify the exception. If the thread pool inside Scala is closed, the abnormal stack information "RejectedExecutionException" is displayed. This abnormal stack information will not be displayed if the thread pool inside Scala is not closed.

The error occurs when the application is successfully run and closed. Therefore, the error will not affect the services.

12.23.8.2.15 What Should I Do If the JDBCServer Process is Mistakenly Killed During a Health Check?

Question

During a health check, if the concurrent statements exceed the threshold of the thread pool, the health check statements fail to be executed, the health check program times out, and the Spark JDBCServer process is killed.

Answer

There are two thread poolsHiveServer2-Handler-Pool and HiveServer2-Background-Pool in the current JDBCServer. The HiveServer2-Handler-Pool is used to connect sessions and the HiveServer2-Background-Pool is used to run Spark SQL statements.

The current health check mechanism establishes a session connection and runs the health check command ***HEALTHCHECK*** in the thread of the session to check the health condition of the Spark JDBCServer. Therefore, one thread must be reserved for the HiveServer2-Handler-Pool respectively to connect sessions and run statements for the health check. Otherwise, the session connection and statement running will fail and the Spark JDBCServer will be killed because it is mistakenly considered unhealthy. For example, if there are 100 threads in the HiveServer2-Handler-Pool respectively, a maximum of 99 sessions can be connected.

12.23.8.2.16 Why No Result Is found When 2016-6-30 Is Set in the Date Field as the Filter Condition?

Question

Why no result is found when 2016-6-30 is set in the date field as the filter condition?

As shown in the following figure, trx_dte_par in the select count (*) from trxfintrx2012 a where trx_dte_par='2016-6-30' statement is a date field. However, no search result is found when the filter condition is where trx_dte_par='2016-6-30'. Search results are found only when the filter condition is where trx_dte_par='2016-06-30'.

Figure 12-62 Example

```
0: jdbc:hive2://ha-cluster/default> select count(*)
0: jdbc:hive2://ha-cluster/default>   from TRXFINTRX2012 a
0: jdbc:hive2://ha-cluster/default>   where trx_dte_par = '2016-6-30';
+------+--+
| _c0  |
+------+--+
| 0    |
+------+--+
1 row selected (0.498 seconds)
0: jdbc:hive2://ha-cluster/default> select count(*)
0: jdbc:hive2://ha-cluster/default>   from TRXFINTRX2012 a
0: jdbc:hive2://ha-cluster/default>   where trx_dte_par = '2016-06-30';
+----------+--+
|   _c0    |
+----------+--+
| 8520808  |
+----------+--+
1 row selected (15.788 seconds)
```

Answer

If a data string of the date type is present in Spark SQL statements, the Spark SQL will search the matching character string without checking the date format. In this case, if the date format in the SQL statement is incorrect, the query will fail. For example, if the data format is yyyy-mm-dd, then no search results matching '2016-6-30' will be found.

12.23.8.2.17 Why Does the "--hivevar" Option I Specified in the Command for Starting spark-beeline Fail to Take Effect?

Question

Why does the ***--hivevar*** option I specified in the command for starting spark-beeline fail to take effect?

In the V100R002C60 version, if I use the ***--hivevar <VAR_NAME>=<var_value>*** option to define a variable in the command for starting spark-beeline, no error is reported in spark-beeline. However, if the variable *<VAR_NAME>* is used in SQL, the variable cannot be parsed and the <VAR_NAME> exception is reported.

For example:

1. Run the following command to start the spark-beeline:

   spark-beeline --hivevar <VAR_NAME>=<var_value>

2. After spark-beeline is started successfully, I run the SQL statements ***DROP TABLE ${VAR_NAME}*** in spark-beeline. The VAR_NAME exception occurs.

Answer

In the V100R002C60 version, the ***--hivevar <VAR_NAME>=<var_value>*** feature of Hive is not supported in Spark because multi-session management function is added. Therefore, the ***--hivevar*** option in the command for starting spark-beeline is invalid.

12.23.8.2.18 Why Does the "Permission denied" Exception Occur When I Create a Temporary Table or View in Spark-beeline?

Question

In normal mode, when I create a temporary table or view in spark-beeline, the error message "Permission denied" is displayed, indicating that I have no permissions on the HDFS directory. The error log information is as follows:

```
org.apache.hadoop.security.AccessControlException Permission denied: user=root, access=EXECUTE,
inode="/tmp/spark/sparkhive-scratch/omm/e579a76f-43ed-4014-8a54-1072c07ceeff/_tmp_space.db/
52db1561-60b0-4e7d-8a25-c2eaa44850a9":omm:hadoop:drwx------
```

Answer

In normal mode, if you run the spark-beeline command as a non-omm user, **root** user for example, without specifying the **-n** parameter, your account is still the root user. After spark-beeline is started, a new HDFS directory is created by JDBCServer. In the current version of DataSight, the user that starts the JDBCServer is **omm**. In versions earlier than DataSight V100R002C30, the user is **root**. Therefore, the owner of the HDFS directory is **omm** and the group is **hadoop**. The HDFS directory is used when you create a temporary table or view in spark-beeline and the user **root** is a common user in HDFS and has no permissions on the directory of user **omm**. As a result, the "Permission denied" exception occurs.

In normal mode, only user **omm** can create a temporary table or view. To solve this problem, you can specify the **-n omm** option for user **omm** when starting spark-beeline. In this way, you have the permissions to perform operations on the HDFS directory.

12.23.8.2.19 Why Is the "Code of method ... grows beyond 64 KB" Error Message Displayed When I Run Complex SQL Statements?

Question

When I run a complex SQL statement, for example, SQL statements with multiple layers of nesting statements and a single layer statement contains a large number of logic clauses such as case when, an error message indicating that the code of a certain method exceeds 64 KB is displayed. The log is as follows:

```
java.util.concurrent.ExecutionException: java.lang.Exception: failed to compile:
org.codehaus.janino.JaninoRuntimeException: Code of method "(Lorg/apache/spark/sql/catalyst/expressions/
GeneratedClass$SpecificUnsafeProjection;Lorg/apache/spark/sql/catalyst/InternalRow;)V" of class
"org.apache.spark.sql.catalyst.expressions.GeneratedClass$SpecificUnsafeProjection" grows beyond 64 KB
```

Answer

If Project Tungsten is enabled, Spark will use codegen method to generate Java code for part of execution plan. However, each function in Java code to be compiled by JDK must be less than 64 KB. If complex SQL statements are run, the function in the Java code generated by codegen may exceed 64 KB, causing compilation failure.

To solve the problem, go to the **spark-defaults.conf** file on the client and set the **spark.sql.codegen.wholeStage** parameter to **false** to disable Project Tungsten.

12.23.8.2.20 Why Is Memory Insufficient if 10 Terabytes of TPCDS Test Suites Are Consecutively Run in Beeline/JDBCServer Mode?

Question

When the driver memory is set to 10 GB and the 10 TB TPCDS test suites are continuously run in Beeline/JDBCServer mode, SQL statements fail to be executed due to insufficient driver memory. Why?

Answer

By default, 1000 UI data records of jobs and stages are reserved in the memory.

The function of overflowing UI data to disks has been added to optimize large clusters. The overflow condition is that the size of UI data in each stage reaches the minimum threshold 5 MB. If the number of tasks in each stage is small, the size of UI data in the stage may not reach the threshold. As a result, the UI data in the stage is cached in the memory until the number of UI data records reaches the upper limit (1000 by default). Only then the old UI data is cleared from the memory.

Therefore, before the old UI data is cleared, the UI data occupies a large amount of memory. As a result, the driver memory is insufficient when 10 terabytes of TPCDS test suites are executed.

Workaround:

- Set **spark.ui.retainedJobs** and **spark.ui.retainedStages** based on service requirements to specify the number of UI data records of jobs and stages to be reserved. For details, see **Table 12-359** in **Common Parameters**.

- If a large amount of UI data of jobs and stages needs to be reserved, increase the memory of the driver by setting the **spark.driver.memory** parameter. For details, see **Table 12-356** in **Common Parameters**.

12.23.8.2.21 Why Are Some Functions Not Available when Another JDBCServer Is Connected?

Question

Scenario 1

I set up permanent functions using the **add jar** statement. After Beeline connects to different JDBCServer or JDBCServer is restarted, I have to run the **add jar** statement again.

Figure 12-63 Error information in scenario 1

```
0: jdbc:hive2://192.168.91.247:23040/default> create function a1 as '                                                    ';
+---------+--+
| result  |
+---------+--+
+---------+--+
No rows selected (0.222 seconds)
0: jdbc:hive2://192.168.91.247:23040/default> SELECT test.a1(array(1, 2, 3), array(2));
+--------+--+
|  _c0   |
+--------+--+
| true   |
+--------+--+
1 row selected (8.282 seconds)
0: jdbc:hive2://192.168.91.247:23040/default> Closing: 0: jdbc:hive2://192.168.91.247:24002,192.168.154.81:24002,192.168.8.27:24002;serviceDiscoveryMode=zooKeep
p=auth-conf;auth=KERBEROS;principal=spark/hadoop.hadoop.com@HADOOP.COM;
100-106-121-140:/opt/hadoopclient # ./spark-beeline
It's running the fi spark-beeline, it calls /opt/hadoopclient/Spark/spark/bin/beeline
and helps to connect to the JDBCServer automatically
Connecting to jdbc:hive2://192.168.91.247:24002,192.168.154.81:24002,192.168.8.27:24002;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=sparkthriftserver;sas
doop.hadoop.com@HADOOP.COM;
2017-06-15 08:17:55,495 | WARN  | Thread-2 | TGT refresh thread time adjusted from : Thu Jun 15 05:59:42 GMT+08:00 2017 to : Thu Jun 15 08:18:55 GMT+08:00 2017
fresh interval (60 seconds) from now. | org.apache.zookeeper.Login$1.run(Login.java:177)
2017-06-15 08:17:56,743 | WARN  | main | Unable to load native-hadoop library for your platform... using builtin-java classes where applicable | org.apache.had
ader.java:62)
2017-06-15 08:17:56,773 | WARN  | TGT Renewer for sparkuser@HADOOP.COM | Exception encountered while running the renewal command. Aborting renew thread. ExitCo
l requested option while renewing credentials
 | org.apache.hadoop.security.UserGroupInformation$1.run(UserGroupInformation.java:946)
Connected to: Spark SQL (version)
Driver: Hive JDBC (version 1.2.1.spark)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.2.1.spark by Apache Hive
[INFO] Unable to bind key for unsupported operation: backward-delete-word
[INFO] Unable to bind key for unsupported operation: backward-delete-word
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
0: jdbc:hive2://192.168.8.27:23040/default> SELECT test.a1(array(1, 2, 3), array(2));
Error: org.apache.spark.SparkException: Unable to load UDF class (state=,code=0)
0: jdbc:hive2://192.168.8.27:23040/default> set role admin;
+-------------+--------+--+
|     key     | value  |
+-------------+--------+--+
| role admin  |        |
+-------------+--------+--+
1 row selected (0.465 seconds)
0: jdbc:hive2://192.168.8.27:23040/default> add jar /home/smartcare-udf-0.0.1-SNAPSHOT.jar;
+---------+--+
| result  |
+---------+--+
| 0       |
+---------+--+
```

Scenario 2

The **show functions** statement can be used to query functions, but not obtain functions. The reason is that connected JDBC node does not contain jar packages of the corresponding path. However, after I add corresponding **.jar** packages, the **show functions** statement can be used to obtain functions.

Figure 12-64 Error information in scenario 2

Answer

Scenario 1

The ***add jar*** statement is used to load jars to the jarClassLoader of the JDBCServer connected currently. The ***add jar*** statement is not shared by different JDBCServer. After the JDBCServer restarts, new jarClassLoader is created. So the add jar statement needs to be run again.

There are two methods to add jar packages: You can run the **spark-sql --jars /opt/test/two_udfs.jar** statement to add the jar package during the startup of the Spark SQL process; or run the **add jar /opt/test/two_udfs.jar** statement to add the jar package after the Spark SQL process is started. Note that the path following the add jar statement can be a local path or an HDFS path.

Scenario 2

The show functions statement is used to obtain all functions in the current database from the external catalog. If functions are used in SQL, thriftJDBC-server loads **.jar** files related to the function.

If **.jar** files do not exist, the function cannot obtain corresponding **.jar** files. Therefore, the corresponding **.jar** files need to be added.

12.23.8.2.22 Why Does Spark2x Have No Access to DataSource Tables Created by Spark1.5?

Question

When Spark2x accesses the DataSource table created by Spark1.5, a message is displayed indicating that schema information cannot be obtained. As a result, the table cannot be accessed. Why?

Answer

- Cause analysis:

  This is because the formats of the DataSource table information stored in Spark2x and Spark1.5 are inconsistent. Spark 1.5 divides schema information into multiple parts and uses **path.park.0** as the key for storage. Spark 1.5 reads information from each part and reassembles the information into complete one. Spark2x directly uses the corresponding key to obtain the corresponding information. In this case, when Spark2x reads the DataSource table created by Spark1.5, the information corresponding to the key cannot be read. As a result, the DataSource table information fails to be parsed.

  When processing Hive tables, Spark2x and Spark1.5 use the same storage mode. Therefore, Spark2x can directly read tables created by Spark1.5.

- Workaround:

  In Spark2x, create a foreign table to point to the actual data in the Spark1.5 table. In this way, the DataSource table created by Spark1.5 can be read in Spark2x. In addition, after Spark1.5 updates data, Spark2x can detect the change. The reverse is also true. In this way, Spark2x can access the DataSource table created by Spark1.5.

12.23.8.2.23 Why Does Spark-beeline Fail to Run and Error Message "Failed to create ThriftService instance" Is Displayed?

Question

Why does "Failed to create ThriftService instance" occur when spark beeline fails to run?

Beeline logs are as follows:

```
Error: Failed to create ThriftService instance (state=,code=0)
Beeline version 1.2.1.spark by Apache Hive
[INFO] Unable to bind key for unsupported operation: backward-delete-word
[INFO] Unable to bind key for unsupported operation: backward-delete-word
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
[INFO] Unable to bind key for unsupported operation: up-history
[INFO] Unable to bind key for unsupported operation: down-history
beeline>
```

In addition, the "Timed out waiting for client to connect" error log is generated on the JDBCServer. The details are as follows:

```
2017-07-12 17:35:11,284 | INFO  | [main] | Will try to open client transport with JDBC Uri:
jdbc:hive2://192.168.101.97:23040/default;principal=spark/hadoop.<System domain name>@<System
domain name>;healthcheck=true;saslQop=auth-conf;auth=KERBEROS;user.principal=spark/hadoop.<System
domain name>@<System domain name>;user.keytab=${BIGDATA_HOME}/FusionInsight_HD_8.1.0.1/install/
FusionInsight-Spark-3.1.1/keytab/spark/JDBCServer/spark.keytab |
org.apache.hive.jdbc.HiveConnection.openTransport(HiveConnection.java:317)
2017-07-12 17:35:11,326 | INFO  | [HiveServer2-Handler-Pool: Thread-92] | Client protocol version:
HIVE_CLI_SERVICE_PROTOCOL_V8 |
org.apache.proxy.service.ThriftCLIProxyService.OpenSession(ThriftCLIProxyService.java:554)
2017-07-12 17:35:49,790 | ERROR | [HiveServer2-Handler-Pool: Thread-113] | Timed out waiting for client
to connect.
```

```
Possible reasons include network issues, errors in remote driver or the cluster has no available resources, etc.
Please check YARN or Spark driver's logs for further information. |
org.apache.proxy.service.client.SparkClientImpl.<init>(SparkClientImpl.java:90)
java.util.concurrent.ExecutionException: java.util.concurrent.TimeoutException: Timed out waiting for
client connection.
 at io.netty.util.concurrent.AbstractFuture.get(AbstractFuture.java:37)
 at org.apache.proxy.service.client.SparkClientImpl.<init>(SparkClientImpl.java:87)
 at org.apache.proxy.service.client.SparkClientFactory.createClient(SparkClientFactory.java:79)
 at org.apache.proxy.service.SparkClientManager.createSparkClient(SparkClientManager.java:145)
 at org.apache.proxy.service.SparkClientManager.createThriftServerInstance(SparkClientManager.java:160)
 at org.apache.proxy.service.ThriftServiceManager.getOrCreateThriftServer(ThriftServiceManager.java:182)
 at org.apache.proxy.service.ThriftCLIProxyService.OpenSession(ThriftCLIProxyService.java:596)
 at org.apache.hive.service.cli.thrift.TCLIService$Processor$OpenSession.getResult(TCLIService.java:1257)
 at org.apache.hive.service.cli.thrift.TCLIService$Processor$OpenSession.getResult(TCLIService.java:1242)
 at org.apache.thrift.ProcessFunction.process(ProcessFunction.java:39)
 at org.apache.thrift.TBaseProcessor.process(TBaseProcessor.java:39)
 at org.apache.hadoop.hive.thrift.HadoopThriftAuthBridge$Server
$TUGIAssumingProcessor.process(HadoopThriftAuthBridge.java:696)
 at org.apache.thrift.server.TThreadPoolServer$WorkerProcess.run(TThreadPoolServer.java:286)
 at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
 at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
 at java.lang.Thread.run(Thread.java:748)
Caused by: java.util.concurrent.TimeoutException: Timed out waiting for client connection.
```

Answer

This problem occurs when the network is unstable. When a timed-out exception occurs in beeline, Spark does not attempt to reconnect to beeline. Therefore, you need to restart spark-beeline for reconnection.

12.23.8.2.24 Why Cannot I Query Newly Inserted Data in an ORC Hive Table Using Spark SQL?

Question

Why cannot I query newly inserted data in an ORC Hive table using Spark SQL? This problem occurs in the following scenarios:

- For partitioned tables and non-partitioned tables, after data is inserted on the Hive client, the latest inserted data cannot be queried using Spark SQL.
- After data is inserted into a partitioned table using Spark SQL, if the partition information remains unchanged, the newly inserted data cannot be queried using Spark SQL.

Answer

To improve Spark performance, ORC metadata is cached. When the ORC table is updated by Hive or another means, the cached metadata remains unchanged, resulting in Spark SQL failing to query the newly inserted data.

For an ORC Hive partition table, if the partition information remains unchanged after data is inserted, the cached metadata is not updated. As a result, the newly inserted data cannot be queried by Spark SQL.

Solution

1. To solve the query problem, update metadata before starting a Spark SQL query.

   REFRESH TABLE *table_name*;

table_name indicates the name of the table to be updated. The table must exist. Otherwise, an error is reported.

When the query statement is executed, the latest inserted data can be obtained.

2. Run the following command to disable Spark optimization when using Spark:

   set spark.sql.hive.convertMetastoreOrc=false;

12.23.8.3 Spark Streaming

12.23.8.3.1 What Can I Do If Spark Streaming Tasks Are Blocked?

Question

After a Spark Streaming task is run and data is input, no processing result is displayed. Open the web page to view the Spark job execution status. The following figure shows that two jobs are waiting to be executed but cannot be executed successfully.

Figure 12-65 Active Jobs

Active Jobs (2)

Job Id	Description	Submitted	Duration	Stages: Succeeded/Total
3	print at test2StreamFromKafka.scala:31	2015/05/25 18:28:55	63.7 h	0/3
2	start at test2StreamFromKafka.scala:34	2015/05/25 18:28:55	63.7 h	0/1

Check the completed jobs. Only two jobs are found, indicating that Spark Streaming does not trigger data computing tasks. (By default, Spark Streaming has two jobs that attempt to run. See the figure below.)

Figure 12-66 Completed Jobs

Completed Jobs (2)

Job Id	Description	Submitted	Duration	Stages: Succeeded/Total
1	print at test2StreamFromKafka.scala:31	2015/05/25 18:28:55	0.7 s	2/2 (1 skipped)
0	start at test2StreamFromKafka.scala:34	2015/05/25 18:28:54	1 s	2/2

Answer

After fault locating, it is found that the number of computing cores of Spark Streaming is less than the number of receivers. As a result, after some receivers are started, no resource is available to run computing tasks. Therefore, the first task keeps waiting and subsequent tasks keep queuing. **Figure 12-65** is an example of two queuing tasks.

To address this problem, it is advised to check whether the number of Spark cores is greater than the number of receivers when two tasks are queuing.

NOTE

Receiver is a permanent Spark job in Spark Streaming. It is common for Spark, but its life cycle is the same as that of a Spark Streaming task and occupies one computing core.

Pay attention to the relationship between the number of cores and the number of receivers in scenarios where default configurations are often used, such as debugging and testing.

12.23.8.3.2 What Should I Pay Attention to When Optimizing Spark Streaming Task Parameters?

Question

When Spark Streaming tasks are running, the data processing performance does not improve significantly as the number of executors increases. What should I pay attention to if I perform parameter optimization?

Answer

When the number of executor cores is 1, comply with the following rules to optimize Spark Streaming running parameters:

- The Spark task processing speed is related to the number of partitions in Kafka. When the number of partitions is less than the specified number of executors, the number of actually used executors is the same as the number of partitions, and other executors will be idle. Therefore, the number of executors must be less than or equal to the number of partitions.
- When data skew occurs on different partitions of Kafka, the executor corresponding to the partition with a large amount of data touches the glass ceiling of data processing. Therefore, when the Producer program is executed, data is sent to each partition on average to improve the processing speed.
- When partition data is evenly distributed, increasing the number of partitions and executors will improve the Spark processing speed. (When the number of partitions is the same as that of executors, the processing speed is the fastest.)
- When partition data is evenly distributed, ensure that the number of partitions is an integer multiple of the number of executors for proper allocation of resources.

12.23.8.3.3 Why Does the Spark Streaming Application Fail to Be Submitted After the Token Validity Period Expires?

Question

Change the validity period of the Kerberos ticket and HDFS token to 5 minutes, set **dfs.namenode.delegation.token.renew-interval** to a value less than 60 seconds, and submit the Spark Streaming application. If the token expires, the error message below is displayed, and the application exits. Why?

```
token (HDFS_DELEGATION_TOKEN token 17410 for spark2x) is expired
```

Answer

- Possible causes:

The credential refresh thread of the ApplicationMaster process uploads the updated credential file to the HDFS based on the *token renew period multiplied by 0.75*.

In the executor process, the credential refresh thread obtains the updated credential file from the HDFS based on the time ratio of the *token renewal period multiplied by 0.8* to update the token in UserGroupInformation, preventing the token from being invalid.

When the credential refresh thread of the executor process detects that the current time is later than the credential file update time (*token renew period x 0.8*), it waits for 1 minute and then obtains the latest credential file from the HDFS to ensure that the AM has stored the updated credential file in the HDFS.

When the value of **dfs.namenode.delegation.token.renew-interval** is less than 60 seconds, the started executor detects that the current time is later than the time when the credential file is updated. One minute later, the executor obtains the latest credential file from the HDFS. However, the token is already invalid, and the task fails to be executed. Then, other executor processes retry within 1 minute. The task also fails to run on other executors. As a result, the executors that fail to run are added to the blacklist. If no executors are available, the application exits.

- Solution:

  In the Spark application scenario, set **dfs.namenode.delegation.token.renew-interval** to a value greater than 80 seconds. For details about the **dfs.namenode.delegation.token.renew-interval** parameter, see **Table 12-428**.

  Table 12-428 Parameter description

Parameter	Description	Default Value
dfs.namenode.delegation.token.renew-interval	This parameter is a server parameter. It specifies the maximum lifetime to renew a token. Unit: milliseconds.	86400000

12.23.8.3.4 Why does Spark Streaming Application Fail to Restart from Checkpoint When It Creates an Input Stream Without Output Logic?

Question

Spark Streaming application creates one input stream without output logic. The application fails to restart from checkpoint and an error will be shown like below:

```
17/04/24 10:13:57 ERROR Utils: Exception encountered
java.lang.NullPointerException
at org.apache.spark.streaming.dstream.DStreamCheckpointData$$anonfun$writeObject$1.apply$mcV
$sp(DStreamCheckpointData.scala:125)
at org.apache.spark.streaming.dstream.DStreamCheckpointData$$anonfun$writeObject
$1.apply(DStreamCheckpointData.scala:123)
at org.apache.spark.streaming.dstream.DStreamCheckpointData$$anonfun$writeObject
```

```
$1.apply(DStreamCheckpointData.scala:123)
at org.apache.spark.util.Utils$.tryOrIOException(Utils.scala:1195)
at
org.apache.spark.streaming.dstream.DStreamCheckpointData.writeObject(DStreamCheckpointData.scala:123
)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at java.io.ObjectStreamClass.invokeWriteObject(ObjectStreamClass.java:1028)
at java.io.ObjectOutputStream.writeSerialData(ObjectOutputStream.java:1496)
at java.io.ObjectOutputStream.writeOrdinaryObject(ObjectOutputStream.java:1432
at java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1178)
at java.io.ObjectOutputStream.defaultWriteFields(ObjectOutputStream.java:1548)
at java.io.ObjectOutputStream.defaultWriteObject(ObjectOutputStream.java:441)
at org.apache.spark.streaming.dstream.DStream$$anonfun$writeObject$1.apply$mcV$sp(DStream.scala:515)
at org.apache.spark.streaming.dstream.DStream$$anonfun$writeObject$1.apply(DStream.scala:510)
at org.apache.spark.streaming.dstream.DStream$$anonfun$writeObject$1.apply(DStream.scala:510)
at org.apache.spark.util.Utils$.tryOrIOException(Utils.scala:1195)
at org.apache.spark.streaming.dstream.DStream.writeObject(DStream.scala:510)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at java.io.ObjectStreamClass.invokeWriteObject(ObjectStreamClass.java:1028)
at java.io.ObjectOutputStream.writeSerialData(ObjectOutputStream.java:1496)
at java.io.ObjectOutputStream.writeOrdinaryObject(ObjectOutputStream.java:1432
at java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1178)
at java.io.ObjectOutputStream.writeArray(ObjectOutputStream.java:1378)
at java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1174)
at java.io.ObjectOutputStream.defaultWriteFields(ObjectOutputStream.java:1548)
at java.io.ObjectOutputStream.writeSerialData(ObjectOutputStream.java:1509)
at java.io.ObjectOutputStream.writeOrdinaryObject(ObjectOutputStream.java:1432
at java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1178)
at java.io.ObjectOutputStream.defaultWriteFields(ObjectOutputStream.java:1548)
at java.io.ObjectOutputStream.defaultWriteObject(ObjectOutputStream.java:441)
at org.apache.spark.streaming.DStreamGraph$$anonfun$writeObject$1.apply$mcV
$sp(DStreamGraph.scala:191)
at org.apache.spark.streaming.DStreamGraph$$anonfun$writeObject$1.apply(DStreamGraph.scala:186)
at org.apache.spark.streaming.DStreamGraph$$anonfun$writeObject$1.apply(DStreamGraph.scala:186)
at org.apache.spark.util.Utils$.tryOrIOException(Utils.scala:1195)
at org.apache.spark.streaming.DStreamGraph.writeObject(DStreamGraph.scala:186
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at java.io.ObjectStreamClass.invokeWriteObject(ObjectStreamClass.java:1028)
at java.io.ObjectOutputStream.writeSerialData(ObjectOutputStream.java:1496)
at java.io.ObjectOutputStream.writeOrdinaryObject(ObjectOutputStream.java:1432
at java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1178)
at java.io.ObjectOutputStream.defaultWriteFields(ObjectOutputStream.java:1548)
at java.io.ObjectOutputStream.writeSerialData(ObjectOutputStream.java:1509)
at java.io.ObjectOutputStream.writeOrdinaryObject(ObjectOutputStream.java:1432
at java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1178)
at java.io.ObjectOutputStream.writeObject(ObjectOutputStream.java:348)
at org.apache.spark.streaming.Checkpoint$$anonfun$serialize$1.apply$mcV$sp(Checkpoint.scala:142)
at org.apache.spark.streaming.Checkpoint$$anonfun$serialize$1.apply(Checkpoint.scala:142)
at org.apache.spark.streaming.Checkpoint$$anonfun$serialize$1.apply(Checkpoint.scala:142)
at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1230)
at org.apache.spark.streaming.Checkpoint$.serialize(Checkpoint.scala:143)
at org.apache.spark.streaming.StreamingContext.validate(StreamingContext.scala:566)
at org.apache.spark.streaming.StreamingContext.liftedTree1$1(StreamingContext.scala:612)
at org.apache.spark.streaming.StreamingContext.start(StreamingContext.scala:611)
at com.spark.test.kafka08LifoTwoInkfk$.main(kafka08LifoTwoInkfk.scala:21)
at com.spark.test.kafka08LifoTwoInkfk.main(kafka08LifoTwoInkfk.scala)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
```

```
at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$
$runMain(SparkSubmit.scala:772)
at org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:183)
at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:208)
at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:123)
at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
```

Answer

When Streaming Context starts, DStream checkpoint object of application should be serialized with application set to checkpoint and Dstream context will be used during this serialization.

Dstream.context is the Dstream which Streaming Context relies on to check reversely from output Stream, set the context one by one. If Spark Streaming application creates one input stream which does not have output logic, there will be no context set for the input stream. 'NullPointerException' will be reported during serialization.

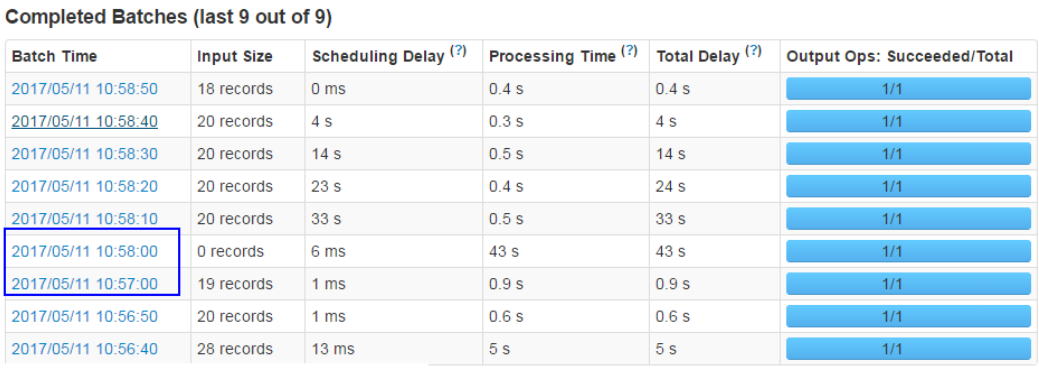
Solution: If there is no input logic for the output stream in the application, delete the input stream in the code or add the relevant output logic for that input stream.

12.23.8.3.5 Why Is the Input Size Corresponding to Batch Time on the Web UI Set to 0 Records When Kafka Is Restarted During Spark Streaming Running?

Question

When the Kafka is restarted during the execution of the Spark Streaming application, the application cannot obtain the topic offset from the Kafka. As a result, the job fails to be generated. As shown in **Figure 12-67**, **2017/05/11 10:57:00-2017/05/11 10:58:00** indicates the Kafka restart time. After the restart is successful at 10:58:00 on May,11,2017, the value of **Input Size** is **0 records**.

Figure 12-67 On the Web UI, the **input size** corresponding to the **batch time** is **0 records**.



Completed Batches (last 9 out of 9)

Batch Time	Input Size	Scheduling Delay (?)	Processing Time (?)	Total Delay (?)	Output Ops: Succeeded/Total
2017/05/11 10:58:50	18 records	0 ms	0.4 s	0.4 s	1/1
2017/05/11 10:58:40	20 records	4 s	0.3 s	4 s	1/1
2017/05/11 10:58:30	20 records	14 s	0.5 s	14 s	1/1
2017/05/11 10:58:20	20 records	23 s	0.4 s	24 s	1/1
2017/05/11 10:58:10	20 records	33 s	0.5 s	33 s	1/1
2017/05/11 10:58:00	0 records	6 ms	43 s	43 s	1/1
2017/05/11 10:57:00	19 records	1 ms	0.9 s	0.9 s	1/1
2017/05/11 10:56:50	20 records	1 ms	0.6 s	0.6 s	1/1
2017/05/11 10:56:40	28 records	13 ms	5 s	5 s	1/1

Answer

After Kafka is restarted, the application supplements the missing RDD between 10:57:00 on May 11, 2017 and 10:58:00 on May 11, 2017 based on the batch time. Although the number of read data records displayed on the UI is **0**, the missing data is processed in the supplemented RDD. Therefore, no data loss occurs.

The data processing mechanism during the Kafka restart period is as follows:

The Spark Streaming application uses the **state** function (for example, **updateStateByKey**). After Kafka is restarted, the Spark Streaming application generates a batch task at 10:58:00 on May 11, 2017. The missing RDD between10:57:00 on May 11, 2017 and 10:58:00 on May 11, 2017 is supplemented based on the batch time (data that is not read in Kafka before Kafka restart, which belongs to the batch before 10:57:00 on May 11, 2017).

12.23.8.4 Why the Job Information Obtained from the restful Interface of an Ended Spark Application Is Incorrect?

Question

The job information obtained from the restful interface of an ended Spark application is incorrect: the value of **numActiveTasks** is negative, as shown in Figure 12-68:

Figure 12-68 job information


```
[ {
  "jobId" : 0,
  "name" : "reduce at SparkPi.scala:36",
  "submissionTime" : "2016-05-28T09:35:34.415GMT",
  "completionTime" : "2016-05-28T09:35:35.686GMT",
  "stageIds" : [ 0 ],
  "status" : "SUCCEEDED",
  "numTasks" : 2,
  "numActiveTasks" : -1,
  "numCompletedTasks" : 2,
  "numSkippedTasks" : 2,
  "numFailedTasks" : 0,
  "numActiveStages" : 0,
  "numCompletedStages" : 1,
  "numSkippedStages" : 0,
  "numFailedStages" : 0
} ]
```


NOTE

numActiveTasks indicates the number of active tasks.

Answer

The job information can be obtained in either of the following methods:

- Set **spark.history.briefInfo.gather=true** and then view the brief JobHistory information.
- Visit the JobHistory2x page of Spark (URL: https://IP:port/api/v1/<appid>/jobs/).

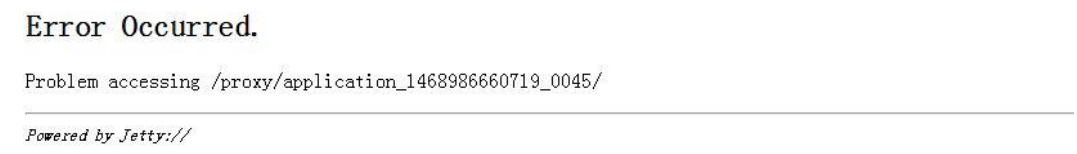
The value of **numActiveTasks** in the job information is calculated from the difference between the number of SparkListenerTaskStart events and the number

of SparkListenerTaskEnd events in the **eventLog** file. If some events are not recorded in the **eventLog** file, the job information obtained from the restful interface is incorrect.

12.23.8.5 Why Cannot I Switch from the Yarn Web UI to the Spark Web UI?

Question

In FusionInsight, the Spark application is run in yarn-client mode on the client. The following error occurs during the switch from the Yarn web UI to the application web UI:



The YARN ResourceManager log shows the following information:

```
2016-07-21 16:35:27,099 | INFO  | Socket Reader #1 for port 8032 | Auth successful for mapred/
hadoop.<System domain name>@<System domain name> (auth:KERBEROS) | Server.java:1388
2016-07-21 16:35:27,105 | INFO  | 1526016381@qtp-1178290888-1015 | admin is accessing unchecked
http://10.120.169.53:23011 which is the app master GUI of
application_1468986660719_0045 owned by spark | WebAppProxyServlet.java:393
2016-07-21 16:36:02,843 | INFO  | Socket Reader #1 for port 8032 | Auth successful for hive/
hadoop.<System domain name>@<System domain name> (auth:KERBEROS) | Server.java:1388
2016-07-21 16:36:02,851 | INFO  | Socket Reader #1 for port 8032 | Auth successful for hive/
hadoop.<System domain name>@<System domain name> (auth:KERBEROS) | Server.java:1388
2016-07-21 16:36:12,163 | WARN  | 1526016381@qtp-1178290888-1015 | /proxy/
application_1468986660719_0045/: java.net.ConnectException: Connection timed out |
Slf4jLog.java:76
2016-07-21 16:37:03,918 | INFO  | Socket Reader #1 for port 8032 | Auth successful for hive/
hadoop.<System domain name>@<System domain name> (auth:KERBEROS) | Server.java:1388
2016-07-21 16:37:03,926 | INFO  | Socket Reader #1 for port 8032 | Auth successful for hive/
hadoop.<System domain name>@<System domain name> (auth:KERBEROS) | Server.java:1388
2016-07-21 16:37:11,956 | INFO  | AsyncDispatcher event handler | Updating application attempt
appattempt_1468986660719_0045_000001 with final state: FINISHING,
and exit status: -1000 | RMAppAttemptImpl.java:1253
```

Answer

On FusionInsight Manager, the IP address of the Yarn service is in the 192 network segment.

In Yarn logs, the IP address of Spark web UI read by Yarn is http://10.120.169.53:23011, which is in the 10 network segment. The IP addresses in the 192 network segment cannot communicate with those in the 10 network segment. As a result, the Spark web UI fails to be accessed.

Solution:

Log in to the client whose IP address is **10.120.169.53** and change the IP address in the **/etc/hosts** file to the IP address in the 192 network segment. Run the Spark application again. The Spark web UI is displayed.

12.23.8.6 What Can I Do If an Error Occurs when I Access the Application Page Because the Application Cached by HistoryServer Is Recycled?

Question

An error occurs when I access a Spark application page on the HistoryServer page.

Check the HistoryServer logs. The "FileNotFound" exception is found. The related logs are as follows:

```
2016-11-22 23:58:03,694 | WARN  | [qtp55429210-232] | /history/application_1479662594976_0001/stages/stage/ | org.sparkproject.jetty.servlet.ServletHandler.doHandle(ServletHandler.java:628)
java.io.FileNotFoundException: ${BIGDATA_HOME}/tmp/spark/jobHistoryTemp/blockmgr-5f1f6aca-2303-4290-9845-88fa94d78480/09/temp_shuffle_11f82aaf-e226-46dc-b1f0-002751557694 (No such file or directory)
```

Answer

If a Spark application with a large number of tasks is run on the HistoryServer page, the memory overflows to disk and files with the **temp_shuffle** prefix are generated.

By default, HistoryServer caches 50 Spark applications (determined by the **spark.history.retainedApplications** configuration item). When the number of Spark applications in the memory exceeds 50, HistoryServer reclaims the first cached Spark application and clears the corresponding **temp_shuffle** file.

When a user is viewing Spark applications to be recycled, the **temp_shuffle** file may not be found. As a result, the current page cannot be accessed.

If the preceding problem occurs, use either of the following methods to solve the problem:

- Access the HistoryServer page of the Spark application again. The correct page information is displayed.
- If more than 50 Spark applications need to be accessed at the same time, increase the value of **spark.history.retainedApplications**.

  Log in to FusionInsight Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Spark2x** > **Configuration**, and click **All Configurations**. In the navigation tree on the left, choose **JobHistory2x** > **GUI**, and set parameters.

  Table 12-429 Parameter description

Parameter	Description	Default Value
spark.history.retainedApplications	Number of Spark applications cached by HistoryServer. When the number of applications to be cached exceeds the value of this parameter, HistoryServer reclaims the first cached Spark application.	50

12.23.8.7 Why Is not an Application Displayed When I Run the Application with the Empty Part File?

Question

When I run an application with an empty part file in HDFS with the log grouping function enabled, why is not the application displayed on the homepage of JobHistory?

Answer

On the JobHistory page, information about applications is updated only with changed sizes of part files in HDFS. If a file is read for the first time, its size is compared with 0. The file is read only when the file size is greater than 0.

When the log grouping function is enabled, if the application you run does not have jobs in running status, the part file is empty. As a result, JobHistory does not read the part file and the application information is not displayed on the JobHistory page. However, if the size of part file is changed later, the application will be displayed on JobHistory.

12.23.8.8 Why Does Spark2x Fail to Export a Table with the Same Field Name?

Question

The following code fails to be executed on spark-shell of Spark2x:

```
val acctId = List(("49562", "Amal", "Derry"), ("00000", "Fred", "Xanadu"))
val rddLeft = sc.makeRDD(acctId)
val dfLeft = rddLeft.toDF("Id", "Name", "City")
//dfLeft.show
val acctCustId = List(("Amal", "49562", "CO"), ("Dave", "99999", "ZZ"))
val rddRight = sc.makeRDD(acctCustId)
val dfRight = rddRight.toDF("Name", "CustId", "State")
//dfRight.show
val dfJoin = dfLeft.join(dfRight, dfLeft("Id") === dfRight("CustId"), "outer")
dfJoin.show
dfJoin.repartition(1).write.format("com.databricks.spark.csv").option("delimiter", "\t").option("header",
"true").option("treatEmptyValuesAsNulls", "true").option("nullValue", "").save("/tmp/outputDir")
```

Answer

In Spark2x, the duplicate field name of the **join** statement is checked. You need to modify the code to ensure that no duplicate field exists in the saved data.

12.23.8.9 Why JRE fatal error after running Spark application multiple times?

Question

Why JRE fatal error after running Spark application multiple times?

Answer

When you run Spark application multiple times, JRE fatal error occurs and this is due to the problem with the Linux Kernel.

To resolve this issue, upgrade the **kernel version to 4.13.9-2.ge7d7106-default**.

12.23.8.10 "This page can't be displayed" Is Displayed When Internet Explorer Fails to Access the Native Spark2x UI

Question

Occasionally, Internet Explorer 9, Explorer 10, or Explorer 11 fails to access the native Spark2x UI.

Symptom

Internet Explorer 9, Explorer 10, or Explorer 11 fails to access the native Spark UI, as shown in the following figure.

Cause

Some Internet Explorer 9, Explorer 10, or Explorer 11versions fail to handle SSL handshake issues, causing access failure.

Solution

Google Chrome 71 and later versions and Firefox browsers 62 and later versions are recommended.

12.23.8.11 How Does Spark2x Access External Cluster Components?

Question

There are two clusters, cluster 1 and cluster 2. How do I use Spark2x in cluster 1 to access HDFS, Hive, HBase, and Kafka components in cluster 2?

Answer

1. Components in two clusters can access each other. However, there are the following restrictions:
   - Only one Hive MetaStore can be accessed. Specifically, Hive MetaStore in cluster 1 and Hive MetaStore in cluster 2 cannot be accessed at the same time.
   - User systems in different clusters are not synchronized. When users access components in another cluster, user permission is determined by

the user configuration of the peer cluster. For example, if user A of cluster 1 does not have the permissions to access the HBase meta table in cluster 1 but user A of cluster 2 can access the HBase meta table in cluster 2, user A of cluster 1 can access the HBase meta table in cluster 2.

- To enable components in a security cluster to communicate with each other across Manager, you need to configure mutual trust.

2. The following describes how to access Hive, HBase, and Kafka components in cluster 2 as user A.

   NOTE

   The following operations are based on the scenario where a user uses the FusionInsight client to submit the Spark2x application. If the user uses the configuration file directory, the user needs to modify the corresponding file in the configuration directory of the application and upload the configuration file to the executor.

   When the HDFS and HBase clients access the server, **hostname** is used to configure the server address. Therefore, the hosts configuration of all nodes to be accessed must be saved in the **/etc/hosts** file on the client. You can add the host of the peer cluster node to the **/etc/hosts** file of the client node in advance.

   - Access Hive metastore: Replace the **hive-site.xml** file in the **conf** directory of the Spark2x client in cluster 1 with the **hive-site.xml** file in the **conf** directory of the Spark2x client in cluster 2.

     After the preceding operations are performed, you can use Spark SQL to access Hive MetaStore. To access Hive table data, you need to perform the operations in **• Access HDFS of two clusters at the same time:** and set **nameservice** of the peer cluster to **LOCATION**.

   - Access HBase of the peer cluster.

     i. Configure the IP addresses and host names of all ZooKeeper nodes and HBase nodes in cluster 2 in the **/etc/hosts** file on the client node of cluster 1.

     ii. Replace the **hbase-site.xml** file in the **conf** directory of the Spark2x client in cluster 1 with the **hbase-site.xml** file in the **conf** directory of the Spark2x client in cluster 2.

   - Access Kafka: Set the address of the Kafka Broker to be accessed to the Kafka Broker address in cluster 2.

   - Access HDFS of two clusters at the same time:

     - Two tokens with the same NameService cannot be obtained at the same time. Therefore, the NameServices of the HDFS in two clusters must be different. For example, one is **hacluster**, and the other is **test**.

       1) Obtain the following configurations from the **hdfs-site.xml** file of cluster2 and add them to the **hdfs-site.xml** file in the **conf** directory of the Spark2x client in cluster1:

          dfs.nameservices.mappings, **dfs.nameservices**, **dfs.namenode.rpc-address.test.***, **dfs.ha.namenodes.test**, and **dfs.client.failover.proxy.provider.test**

          The following is an example:

          ```
          <property>
          <name>dfs.nameservices.mappings</name>
          <value>[{"name":"hacluster","roleInstances":["14","15"]},
          ```

```
{"name":"test","roleInstances":["16","17"]}]</value>
</property>
<property>
<name>dfs.nameservices</name>
<value>hacluster,test</value>
</property>
<property>
<name>dfs.namenode.rpc-address.test.16</name>
<value>192.168.0.1:8020</value>
</property>
<property>
<name>dfs.namenode.rpc-address.test.17</name>
<value>192.168.0.2:8020</value>
</property>
<property>
<name>dfs.ha.namenodes.test</name>
<value>16,17</value>
</property>
<property>
<name>dfs.client.failover.proxy.provider.test</name>
<value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider
</value>
</property>
```

2) Modify **spark.yarn.extra.hadoopFileSystems = hdfs://test and spark.hadoop.hdfs.externalToken.enable = true** in the **spark-defaults.conf** configuration file under the **conf** directory on the Spark client of cluster 1.

```
spark.yarn.extra.hadoopFileSystems = hdfs://test
spark.hadoop.hdfs.externalToken.enable = true
```

3) In the application submission command, add the **--keytab** and **--principal** parameters and set them to the user who submits the task in cluster1.

4) Use the Spark client of cluster1 to submit the application. Then, the two HDFS services can be accessed at the same time.

– Access HBase of two clusters at the same time:

i. Modify **spark.hadoop.hbase.externalToken.enable = true** in the **spark-defaults.conf** configuration file under the **conf** directory on the Spark client of cluster 1.

```
spark.hadoop.hbase.externalToken.enable = true
```

ii. When accessing HBase, you need to use the configuration file of the corresponding cluster to create a **Configuration** object for creating a **Connection** object.

iii. In an MRS cluster, tokens of multiple HBase services can be obtained at the same time to solve the problem that the executor cannot access HBase. The method is as follows:

Assume that you need to access HBase of the current cluster and HBase of cluster2. Save the **hbase-site.xml** file of cluster2 in a compressed package named **external_hbase_conf*****, and use **--archives** to specify the compressed package when submitting the command.

12.23.8.12 Why Does the Foreign Table Query Fail When Multiple Foreign Tables Are Created in the Same Directory?

Question

Assume there is a data file path named **/test_data_path**. User A creates a foreign table named **tableA** for the directory, and user B creates a foreign table named **tableB** for the directory. When user B performs the insert operation on **tableB**, user A fails to query data using **tableA** and the error "Permission denied" is displayed.

Answer

After user B performs the insert operation on **tableB**, a new data file is generated in the foreign table path and the file belongs to user B. When user A queries data using **tableA**, all files in the foreign table directory are read. In this case, the query fails because user A does not have the read permissions on the file generated by user B.

This problem also occurs in other scenarios. For example, the **inset overwrite** operation will also duplicate other table files in this directory.

Due to the Spark SQL implementation mechanism, check restrictions in this scenario will lead to inconsistency and performance deterioration. Therefore, no restriction is added in this scenario, and this method is not recommended.

12.23.8.13 What Should I Do If the Native Page of an Application of Spark2x JobHistory Fails to Display During Access to the Page

Question

After a Spark application that contains a job with millions of tasks. After the application creation is complete, if you access the native page of the application in JobHistory, the native page of the application can be displayed after a long time. If the native page cannot be displayed within 10 minutes, Error information will be generated for the Proxy.

Figure 12-69 Error information example

Answer

When you switch to the native page of an application on the JobHistory page, JobHistory needs to play back the event log of the application. If the application contains a large number of event logs, the playback takes a long time and the browser takes a long time to navigate you to the native page.

The current browser uses the HTTPd as the proxy to access the JobHistory native page. The proxy timeout duration is 10 minutes. Therefore, if the JobHistory

cannot parse the event log and return the result within 10 minutes, the HTTPd automatically returns the proxy error information to the browser.

Solution

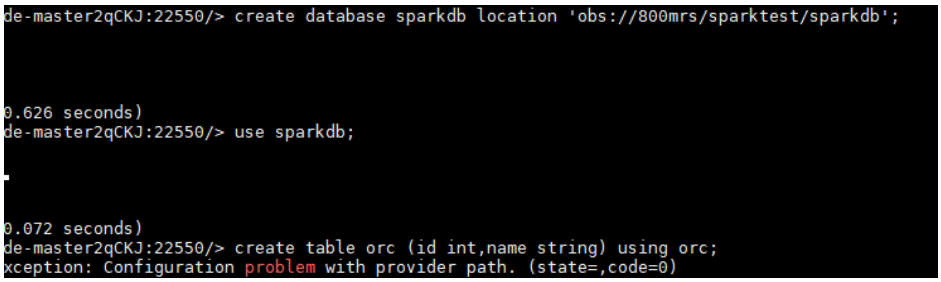
The local disk cache function is enabled on the JobHistory. When a user accesses an application, the event log of the application is cached on the local disk. In this case, the response speed can be greatly accelerated for the second access. Therefore, in this case, you only need to wait for a while and then access the link again. For the second time, you do not need to wait for a long time.

12.23.8.14 Why Do I Fail to Create a Table in the Specified Location on OBS After Logging to spark-beeline?

Question

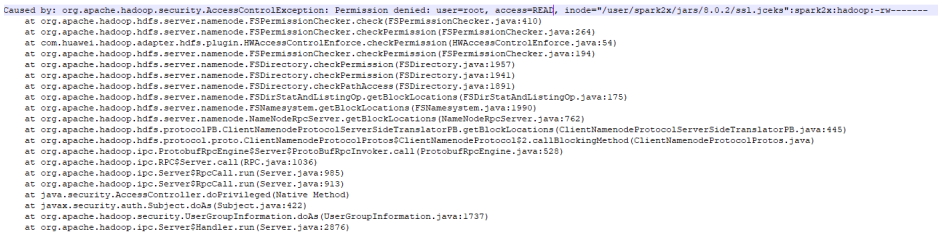
When the OBS ECS/BMS image cluster is connected, after spark-beeline is logged in, an error is reported when a location is specified to create a table on OBS.

Figure 12-70 Error message



Answer

The permission on the **ssl.jceks** file in HDFS is insufficient. As a result, the table fails to be created.



Solution

1. Log in to the node where Spark2x resides as user **omm** and run the following command:

   vi ${BIGDATA_HOME}/FusionInsight_Spark2x_8.1.0.1/install/FusionInsight-Spark2x-3.1.1/spark/sbin/fake_prestart.sh

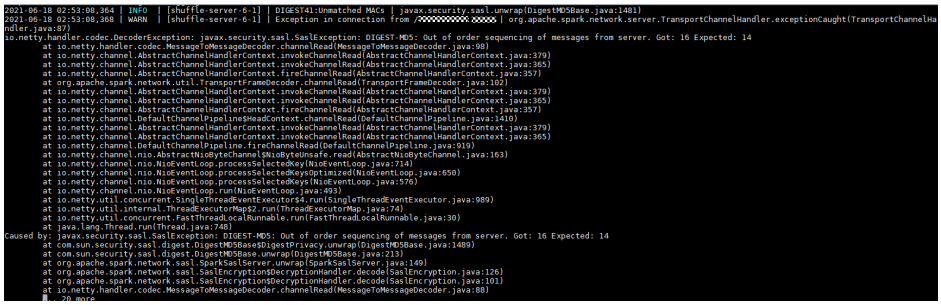
2. Change **eval "${hdfsCmd}" -chmod 600 "${InnerHdfsDir}"/ssl.jceks >> "${PRESTART_LOG}" 2>&1** to **eval "${hdfsCmd}" -chmod 644 "${InnerHdfsDir}"/ssl.jceks >> "${PRESTART_LOG}" 2>&1**.

3. Restart the SparkResource instance.

12.23.8.15 Spark Shuffle Exception Handling

Question

In some scenarios, the following exception occurs in the Spark shuffle phase:



Solution

For JDBC:

Log in to FusionInsight Manager, change the value of the JDBCServer parameter **spark.authenticate.enableSaslEncryption** to **false**, and restart the corresponding instance.

For client jobs:

When the client submits the application, change the value of **spark.authenticate.enableSaslEncryption** in the **spark-defaults.conf** file to **false**.

12.24 Using Sqoop

12.24.1 Using Sqoop from Scratch

Sqoop is an open-source tool for transferring data between Hadoop (Hive) and traditional databases (such as MySQL and PostgreSQL). It can transfer data from a relational database (such as MySQL, Oracle, and PostgreSQL) to HDFS of Hadoop and the other way around.

Prerequisites

- You have selected the Sqoop component when creating a cluster of MRS 3.1.0 or later.
- You have installed the client. For details, see **Installing a Client (Version 3.x or Later)**. For example, the installation directory of the client is **/opt/client**. The client directory in the following operations is an example. Change it to the actual installation directory.

Exporting Data From HDFS to MySQL Using the sqoop export Command

Step 1 Log in to the node where the client is located.

Step 2 Run the following command to initialize environment variables:

source /opt/client/bigdata_env

Step 3 Run the following command to operate the Sqoop client:

sqoop export --connect jdbc:mysql://10.100.231.134:3306/test --username root --password xxxxxx --table component13 -export-dir hdfs://hacluster/user/hive/warehouse/component_test3 --fields-terminated-by ',' -m 1

Table 12-430 Parameter description

Parameter	Description
-direct	Imports data to a relational database using a database import tool, for example, mysqlimport of MySQL, more efficient than the JDBC connection mode.
-export-dir <dir>	Specifies the source directory for storing data in the HDFS.
-m or -num-mappers <n>	Starts *n* (4 by default) maps to import data concurrently. The value cannot be greater than the maximum number of maps in a cluster.
-table <table-name>	Specifies the relational database table to be imported.
-update-key <col-name>	Specifies the column used for updating the existing data in a relational database.
-update-mode <mode>	Specifies how updates are performed. The value can be **updateonly** or **allowinsert**. This parameter is used only when the relational data table does not contain the data record to be imported. For example, if the HDFS data to be imported to the destination table contains a data record **id=1** and the table contains an existing data record **id=2**, the update will fail.
-input-null-string <null-string>	This parameter is optional. If it is not specified, **null** will be used.
-input-null-non-string <null-string>	This parameter is optional. If it is not specified, **null** will be used.

Parameter	Description
-staging-table <staging-table-name>	Creates a table with the same data structure as the destination table for storing data before it is imported to the destination table. This parameter ensures the transaction security when data is imported to a relational database table. Due to multiple transactions during an import, this parameter can prevent other transactions from being affected when one transaction fails. For example, the imported data is incorrect or duplicate records exist.
-clear-staging-table	Clears data in the staging table before data is imported if the staging-table is not empty.

----End

Importing Data from MySQL to Hive Using the sqoop import Command

Step 1 Log in to the node where the client is located.

Step 2 Run the following command to initialize environment variables:

source /opt/client/bigdata_env

Step 3 Run the following command to operate the Sqoop client:

sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password xxxxxx --table component --hive-import --hive-table component_test2 --delete-target-dir --fields-terminated-by "," -m 1 --as-textfile

Table 12-431 Parameter description

Parameter	Description
-append	Appends data to an existing dataset in the HDFS. Once this parameter is used, Sqoop imports data to a temporary directory, renames the temporary file where the data is stored, and moves the file to a formal directory to avoid duplicate file names in the directory.
-as-avrodatafile	Imports data to a data file in the Avro format.
-as-sequencefile	Imports data to a sequence file.
-as-textfile	Import data to a text file. After the text file is generated, you can run SQL statements in Hive to query the result.

Parameter	Description
-boundary-query <statement>	Specifies the SQL statement for performing boundary query. Before importing data, use a SQL statement to obtain a result set and import the data in the result set. The data format can be **-boundary-query 'select id,creationdate from person where id = 3'** (indicating a data record whose ID is 3) or **select min(<split-by>), max(<split-by>) from <table name>**. The fields to be queried cannot contain fields whose data type is string. Otherwise, the error message "java.sql.SQLException: Invalid value for getLong()" is displayed.
-columns<col,col,col...>	Specifies the fields to be imported. The format is ***-Column id,Username***.
-direct	Imports data to a relational database using a database import tool, for example, mysqlimport of MySQL, more efficient than the JDBC connection mode.
-direct-split-size	Splits the imported streams by byte. Especially when data is imported from PostgreSQL using the direct mode, a file that reaches the specified size can be divided into several independent files.
-inline-lob-limit	Sets the maximum value of an inline LOB.
-m or -num-mappers	Starts *n* (4 by default) maps to import data concurrently. The value cannot be greater than the maximum number of maps in a cluster.
-query, -e<statement>	Imports data from the query result. To use this parameter, you must specify the **-target-dir** and **-hive-table** parameters and use the query statement containing the WHERE clause as well as $CONDITIONS. Example: **-query'select * from person where $CONDITIONS' -target-dir /user/hive/warehouse/person -hive-table person**
-split-by<column-name>	Specifies the column of a table used to split work units. Generally, the column name is followed by the primary key ID.
-table <table-name>	Specifies the relational database table from which data is obtained.
-target-dir <dir>	Specifies the HDFS path.
-warehouse-dir <dir>	Specifies the directory for storing data to be imported. This parameter is applicable when data is imported to HDFS but cannot be used when you import data to Hive directories. This parameter cannot be used together with **-target-dir**.

Parameter	Description
-where	Specifies the WHERE clause when data is imported from a relational database, for example, **-where 'id = 2'**.
-z,-compress	Compresses sequence, text, and Avro data files using the GZIP compression algorithm. Data is not compressed by default.
–compression-codec	Specifies the Hadoop compression codec. GZIP is used by default.
–null-string <null-string>	Specifies the string to be interpreted as **NULL** for string columns.
–null-non-string<null-string>	Specifies the string to be interpreted as null for non-string columns. If this parameter is not specified, **NULL** will be used.
-check-column (col)	Specifies the column for checking incremental data import, for example, **id**.
-incremental (mode) append or last modified	Incrementally imports data. **append**: appends records, for example, appending records that are greater than the value specified by **last-value**. **lastmodified**: appends data that is modified after the date specified by **last-value**.
-last-value (value)	Specifies the maximum value (greater than the specified value) of the column after the last import. This parameter can be set as required.

----End

Sqoop Usage Example

- Importing data from MySQL to HDFS using the **sqoop import** command

  sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* **--query 'SELECT * FROM component where $CONDITIONS and component_id ="MRS 1.0_002"' --target-dir /tmp/component_test --delete-target-dir --fields-terminated-by "," -m 1 --as-textfile**

- Exporting data from OBS to MySQL using the **sqoop export** command

  sqoop export --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* **--table component14 -export-dir obs://obs-file-bucket/xx/part-m-00000 --fields-terminated-by ',' -m 1**

- Importing data from MySQL to OBS using the **sqoop import** command

  sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* **--table component --target-dir obs://obs-file-bucket/xx --delete-target-dir --fields-terminated-by "," -m 1 --as-textfile**

- Importing data from MySQL to OBS tables outside Hive

  sqoop import --connect jdbc:mysql://10.100.231.134:3306/test --username root --password *xxx* --table component --hive-import --hive-table component_test01 --fields-terminated-by "," -m 1 --as-textfile

12.24.2 Adapting Sqoop 1.4.7 to MRS 3.*x* Clusters

Sqoop is a tool designed for efficiently transmitting a large amount of data between Apache Hadoop and structured databases (such as relational databases). Customers need to use Sqoop to migrate data in MRS. However, MRS of an earlier version does not provide Sqoop. This section describes how to install and use Sqoop. In MRS 3.1.0 or later, you can select the Sqoop component during cluster creation.

Prerequisites

The MRS client and the JDK environment have been installed.

```
[root@node-masterlfKEj bin]# echo $JAVA_HOME
/opt/Bigdata/client/JDK/jdk1.8.0_242
```

Procedure

Step 1 Download the open-source **sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz** package.

Step 2 Save the downloaded package to the **/opt/Bigdata/client** directory on the node where the MRS client is installed and decompress it.

tar zxvf sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz

Step 3 Download the MySQL JDBC driver **mysql-connector-java-*xxx*.jar** from the MySQL official website. For details about how to select the MySQL JDBC driver, see the following table.

Table 12-432 Version information

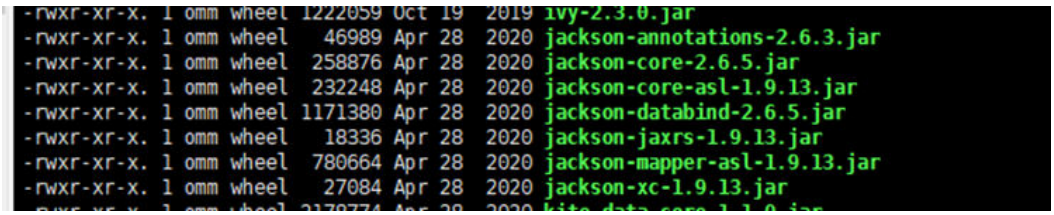
JDBC Driver Version	MySQL Version
Connector/J 5.1	MySQL 4.1, MySQL 5.0, MySQL 5.1, and MySQL 6.0 alpha
Connector/J 5.0	MySQL 4.1, MySQL 5.0 servers, and distributed transaction (XA)
Connector/J 3.1	MySQL 4.1, MySQL 5.0 servers, and MySQL 5.0 except distributed transaction (XA)
Connector/J 3.0	MySQL 3.x and MySQL 4.1

Step 4 Put the MySQL driver package in the **/opt/Bigdata/client/sqoop-1.4.7.bin__hadoop-2.6.0/lib** directory of Sqoop and modify the owner group and permission of the JAR package. For details, see the owner group and permission of **omm:wheel** and **755** in **Figure 12-71**.

Figure 12-71 Owner group and permission of the MySQL driver package

Step 5 Replace the JAR package in the **lib** directory of Sqoop with that starting with **jackson** in the **lib** directory of Hive on the MRS client, for example, **/opt/Bigdata/client/Hive/Beeline/lib**.

Figure 12-72 JAR package starting with **jackson**



Step 6 Copy the **jline** package from the **/opt/Bigdata/client/Hive/Beeline/lib** directory of the MRS Hive client to the **lib** directory of Sqoop.

Step 7 Run the **vim $JAVA_HOME/jre/lib/security/java.policy** command to add the following configuration:

permission javax.management.MBeanTrustPermission "register";

Step 8 Run the following commands to go to the **conf** directory of the Sqoop and add the configuration items of variables:

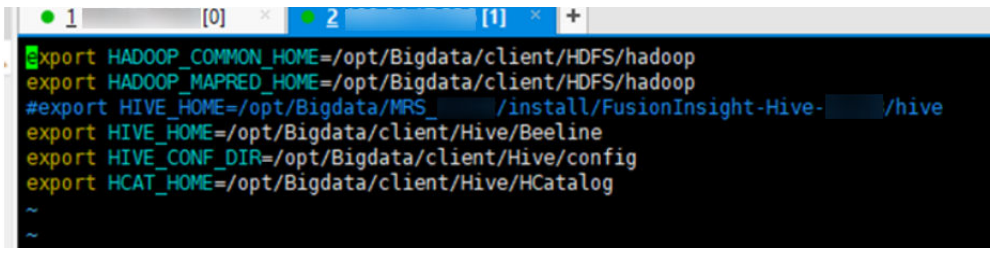
cd /opt/Bigdata/client/sqoop-1.4.7.bin__hadoop-2.6.0/conf

cp sqoop-env-template.sh sqoop-env.sh

Step 9 Run the **vim sqoop-env.sh** command to set the environment variables of Sqoop. Change the Hadoop and Hive directories as required.

```
export HADOOP_COMMON_HOME=/opt/Bigdata/client/HDFS/hadoop
export HADOOP_MAPRED_HOME=/opt/Bigdata/client/HDFS/hadoop
export HIVE_HOME=/opt/Bigdata/MRS_1.9.X/install/FusionInsight-Hive-3.1.0/hive (Enter the actual path.)
export HIVE_CONF_DIR=/opt/Bigdata/client/Hive/config
export HCAT_HOME=/opt/Bigdata/client/Hive/HCatalog
```

Figure 12-73 Setting environment variables of Sqoop



Step 10 Build the sqoop script. For example:

```
/opt/Bigdata/FusionInsight_Current/1_19_SqoopClient/install/FusionInsight-Sqoop-1.4.7/bin/sqoop import
--connect jdbc:mysql://192.168.0.183:3306/test
--driver com.mysql.jdbc.Driver
--username 'root'
--password 'xxx'
--query "SELECT id, name  FROM tbtest WHERE \$CONDITIONS"
--hcatalog-database default
```

```
--hcatalog-table test
--num-mappers 1
```

----End

12.24.3 Common Sqoop Commands and Parameters

Common Sqoop commands

Table 12-433 Common Sqoop commands

Command	Description
import	Imports data to a cluster.
export	Exports data of a cluster.
codegen	Obtains data from a table in the database to generate a Java file and compress the file.
create-hive-table	Creates a Hive table.
eval	Executes a SQL statement and view the result.
import-all-tables	Imports all tables in a database to HDFS.
job	Generates a Sqoop job.
list-databases	Lists database names.
list-tables	List table names.
merge	Merges data in different HDFS directories and saves the data to a specified directory.
metastore	Starts the metadata database to record the metadata of a Sqoop job.
help	Prints help information.
version	Prints the version information.

Common Parameters

Table 12-434 Common parameters

Category	Parameter	Description
Parameters for database connection	--connect	Specifies the URL for connecting to a relational database.
	--connection-manager	Specifies the connection manager class.

Category	Parameter	Description
	--driver jdbc	Specifies the driver package for database connection.
	--help	Prints help information.
	--password	Specifies the password for connecting to a database.
	--username	Specifies the username for connecting to a database.
	--verbose	Prints detailed information on the console.
import parameters	--fields-terminated-by	Specifies the field delimiter, which must be the same as that in a Hive table or HDFS file.
	--lines-terminated-by	Specifies the line delimiter, which must be the same as that in a Hive table or HDFS file.
	--mysql-delimiters	Specifies the default delimiter settings of MySQL.
export parameters	--input-fields-terminated-by	Specifies the field delimiter.
	--input-lines-terminated-by	Specifies the line delimiter.
Hive parameters	--hive-delims-replacement	Replaces characters such as **\r** and **\n** in data with user-defined characters.
	--hive-drop-import-delims	Removes characters such as **\r** and **\n** when data is imported to Hive.
	--map-column-hive	Specifies the data type of fields during the generation of a Hive table.
	--hive-partition-key	Creates a partition.
	--hive-partition-value	Imports data to a specified partition of a database.
	--hive-home	Specifies the installation directory for Hive.
	--hive-import	Specifies that data is imported from a relational database to Hive.
	--hive-overwrite	Overwrites existing Hive data.

Category	Parameter	Description
	--create-hive-table	Creates a Hive table. The default value is **false**. A destination table will be created if it does not exist.
	--hive-table	Specifies a Hive table to which data is to be imported.
	--table	Specifies the relational database table.
	--columns	Specifies the fields of a relational data table to be imported.
	--query	Specifies the query statement for importing the query result.
HCatalog parameters	--hcatalog-database	Specifies a Hive database and imports data to it using HCatalog.
	--hcatalog-table	Specifies a Hive table and imports data to it using HCatalog.
Others	-m or --num-mappers	Specifies the number of map tasks used by a Sqoop job.
	--split-by	Specifies the column based on which Sqoop splits work units. This parameter is used together with **-m**.
	--target-dir	Specifies the temporary directory of HDFS.
	--null-string string	Specifies the string to be written for a null value for string columns.
	--null-non-string	Specifies the string to be written for a null value for non-string columns.
	--check-column	Specifies the column for determining incremental data import.
	--incremental append or lastmodified	Incrementally imports data. **append**: appends records, for example, appending records that are greater than the value specified by **last-value**. **lastmodified**: appends data that is modified after the date specified by **last-value**.
	--last-value	Specifies the last value of the check column from the previous import.
	--input-null-string	Specifies the string to be interpreted as **NULL** for string columns.

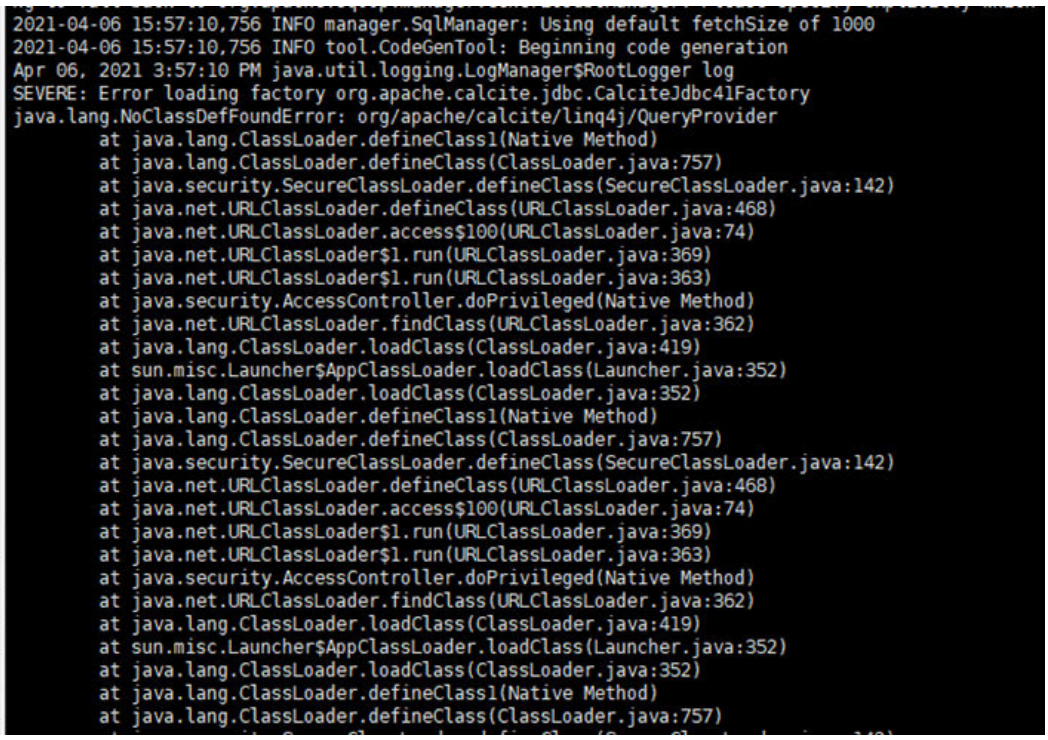
Category	Parameter	Description
	--input-null-non-string	Specifies the string to be interpreted as null for non-string columns. If this parameter is not specified, **NULL** will be used.

12.24.4 Common Issues About Sqoop

12.24.4.1 What Should I Do If Class QueryProvider Is Unavailable?

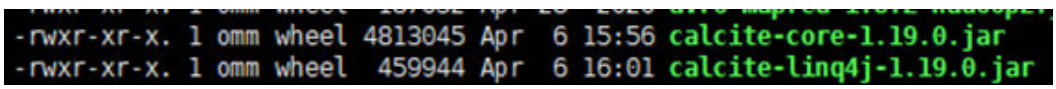
Question

What should I do if the QueryProvider class is unavailable?



Answer

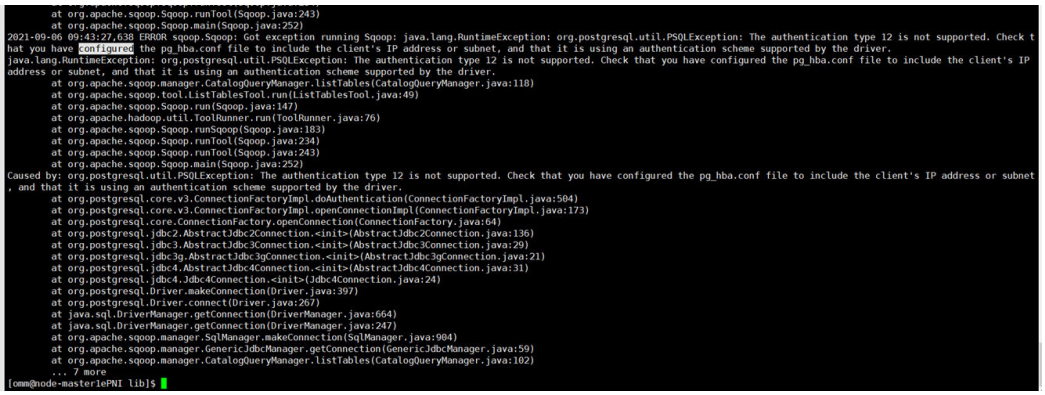
Search for the MRS client directory and save the following JAR packages to the **lib** directory of Sqoop.



12.24.4.2 What Should I Do If PostgreSQL or GaussDB Failed to Be Connected?

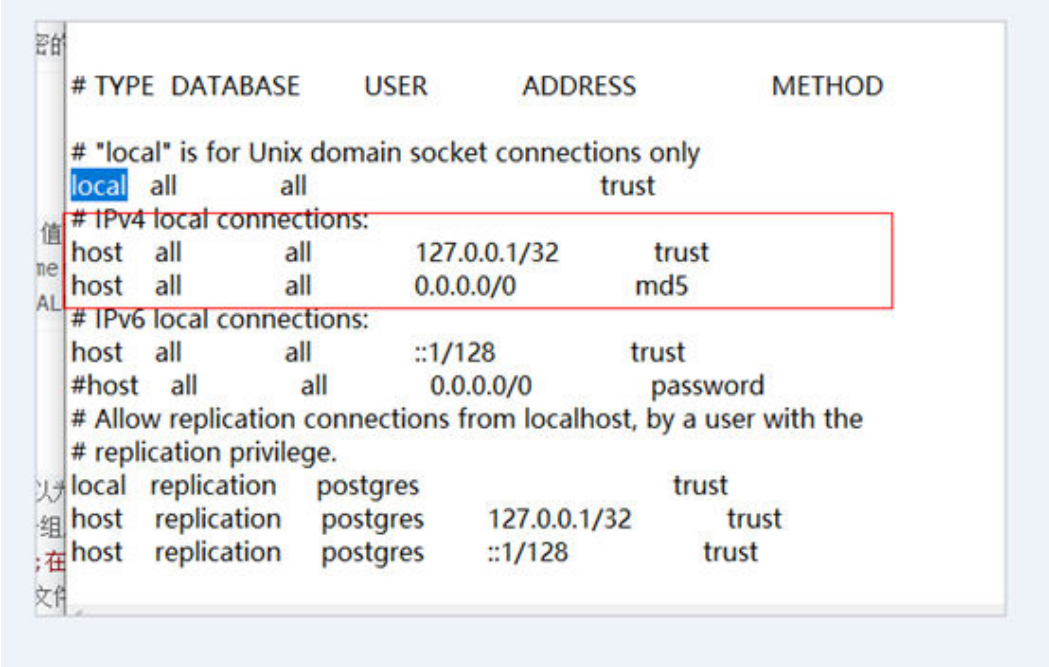
Question

What should I do if PostgreSQL or GaussDB failed to be connected?



Answer

Modify the **pg_hba.conf** file of the database and change the value of **ADDRESS** to the IP address of the node where Sqoop is located.



12.24.4.3 What Should I Do If Data Failed to Be Synchronized to a Hive Table on the OBS Using hive-table?

Question

What should I do if data failed to be synchronized to a Hive table on the OBS using hive-table?

```
2021-09-03 16:28:11,611 ERROR tools.DistCp: XAttrs not supported on at least one file system:
org.apache.hadoop.tools.CopyListing$XAttrsNotSupportedException: XAttrs not supported for file system:
obs://fdd-fs
        at org.apache.hadoop.tools.util.DistCpUtils.checkFileSystemXAttrSupport(DistCpUtils.java:555)
        at org.apache.hadoop.tools.DistCp.configureOutputFormat(DistCp.java:341)
        at org.apache.hadoop.tools.DistCp.createJob(DistCp.java:308)
        at org.apache.hadoop.tools.DistCp.createAndSubmitJob(DistCp.java:218)
        at org.apache.hadoop.tools.DistCp.execute(DistCp.java:197)
        at org.apache.hadoop.tools.DistCp.run(DistCp.java:155)
```

Answer

Change **-hive-table** to **-hcatalog-table**.

12.24.4.4 What Should I Do If Data Failed to Be Synchronized to an ORC or Parquet Table Using hive-table?

Question

What should I do if data failed to be synchronized to the ORC or parquet table using hive-table and error message that contains the **kite-sdk** package name is displayed?

Answer

Change **-hive-table** to **-hcatalog-table**.

12.24.4.5 What Should I Do If Data Failed to Be Synchronized Using hive-table?

Question

What should I do if data failed to be synchronized using hive-table?

```
        at org.apache.hadoop.hive.ql.metadata.Hive.registerAllFunctionsOnce(Hive.java:2907) [hive-exec-3.1.0-RW-ei-312001-SNAPSHOT.jar:3.1.0-RW-ei-312001-SNAPSHOT]
        ... 41 more
14:41:42.891 [bfe1438c-07bb-43fd-91d9-910a348f6e91 main] ERROR org.apache.hadoop.hive.metastore.ObjectStore - Version information not found in metastore. The process will exit.
14:41:42.892 [bfe1438c-07bb-43fd-91d9-910a348f6e91 main] ERROR org.apache.hadoop.hive.metastore.RetryingHMSHandler - ExitSecurityException
        at org.apache.sqoop.util.SubprocessSecurityManager.checkExit(SubprocessSecurityManager.java:83)
        at java.lang.Runtime.exit(Runtime.java:107)
        at java.lang.System.exit(System.java:973)
        at org.apache.hadoop.hive.metastore.ObjectStore.checkSchema(ObjectStore.java:9555)
        at org.apache.hadoop.hive.metastore.ObjectStore.verifySchema(ObjectStore.java:9531)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.hadoop.hive.metastore.RawStoreProxy.invoke(RawStoreProxy.java:97)
        at com.sun.proxy.$Proxy37.verifySchema(Unknown Source)
        at org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMSForConf(HiveMetaStore.java:903)
        at org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:895)
        at org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.java:978)
        at org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:585)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.hadoop.hive.metastore.RetryingHMSHandler.invokeInternal(RetryingHMSHandler.java:148)
        at org.apache.hadoop.hive.metastore.RetryingHMSHandler.invoke(RetryingHMSHandler.java:109)
        at org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:81)
        at org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:94)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.newRetryingHMSHandler(HiveMetaStore.java:9683)
        at org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:185)
        at org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient.<init>(SessionHiveMetaStoreClient.java:96)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
        at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
        at org.apache.hadoop.hive.metastore.utils.JavaUtils.newInstance(JavaUtils.java:84)
        at org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:97)
```

Answer

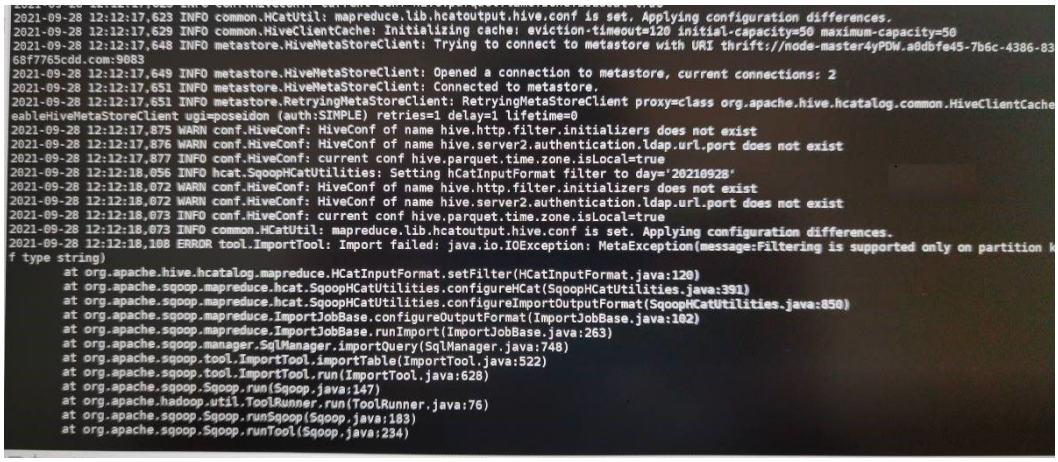
Add the following content to the **hive-site.xml** file.

```
<property>
<name>hive.metastore.schema.verification</name>
<value>false</value>
</property>
```

12.24.4.6 What Should I Do If Data Failed to Be Synchronized to a Hive Parquet Table Using HCatalog?

Question

When the partition fields in a Hive parquet table are not of the string type, data in the table can be synchronized only using HCatalog. What should I do if the following error message is displayed during data synchronization?



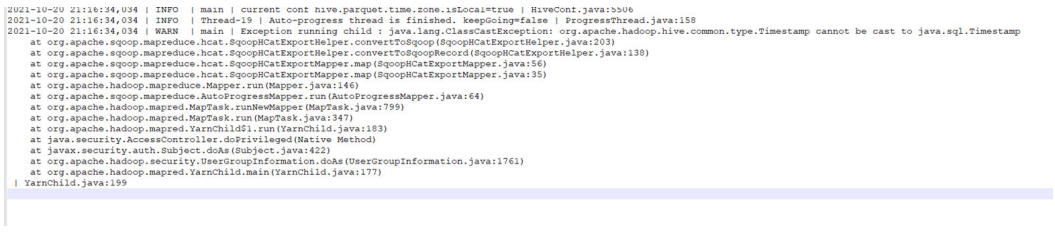
Answer

1. Delete the restricted code in the **SqoopHCatUtilities** class of Sqoop.
2. Change the value of the **hive.metastore.integral.jdo.pushdown** parameter in the **hive-site.xml** file on the Hive client to **true**.

12.24.4.7 What Should I Do If the Data Type of Fields timestamp and data Is Incorrect During Data Synchronization Between Hive and MySQL?

Question

What should I do if the data type of fields timestamp and data is incorrect during data synchronization between Hive and MySQL?



Answer

- Forcibly convert the data type of the timestamp field in the Sqoop source package to be the same as that in Hive.
- Change the data type of the timestamp field in Hive to String.

12.25 Using Storm

12.25.1 Using Storm from Scratch

You can submit and delete Storm topologies on the MRS cluster client.

Prerequisites

The MRS cluster client has been installed, for example, in the **/opt/hadoopclient** directory. The client directory in the following operations is only an example. Change it based on the actual installation directory onsite.

Procedure

Step 1 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see .

Step 2 Run the following command to switch to the client directory, for example, **/opt/hadoopclient**:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 For clusters with Kerberos authentication enabled, run the following command to authenticate the user. For clusters with Kerberos authentication disabled, skip this step.

kinit *Storm user*

Step 5 Run the following command to submit the Storm topology:

storm jar *Path of the topology package Class name of the topology Main method Topology name*

If the following information is displayed, the topology is submitted successfully.

```
Finished submitting topology: topo1
```

Step 6 Run the following command to query Storm topologies. For clusters with Kerberos authentication enabled, only users in the **stormadmin** or **storm** group can query all topologies.

storm list

Step 7 Run the following command to delete a Storm topology.

storm kill *Topology name*

----End

12.25.2 Using the Storm Client

Scenario

This section describes how to use the Storm client in an O&M scenario or service scenario.

Prerequisites

- You have installed the client. For example, the installation directory is **/opt/hadoopclient**.
- Service component users are created by the administrator as required. In security mode, machine-machine users have downloaded the keytab file. A human-machine user must change the password upon the first login. (Not involved in normal mode)

Procedure

Step 1 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see **Using an MRS Client**.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If multiple Storm instances are installed, run the following command to load the environment variables of a specific instance when running the Storm command to submit the topology. Otherwise, skip this step. The following command uses the instance Storm-2 as an example.

source Storm-2/component_env

Step 5 Run the following command to perform user authentication (skip this step in normal mode):

kinit *Component service user*

Step 6 Run the following command to perform operations on the client:

For example, run the following command:

- **cql**
- **storm**

NOTE

A Storm client cannot be connected to secure and non-secure ZooKeepers at the same time.

----End

12.25.3 Submitting Storm Topologies on the Client

Scenario

You can submit Storm topologies on the cluster client to continuously process stream data. For clusters with Kerberos authentication enabled, users who submit topologies must be members of the **stormadmin** or **storm** group.

Prerequisites

The client has been updated.

Procedure

Step 1 Prepare the client based on service requirements. Log in to the node where the client is installed.

Log in to the node where the client is installed. For details, see **Using an MRS Client**.

Step 2 Run the following command to set the permissions on the topology JAR file:

For example, run the following command to change the permissions on **/opt/storm/topology.jar**:

chmod 600 /opt/storm/topology.jar

Step 3 Run the following command to switch to the client directory, for example, **/opt/client**.

cd /opt/client

Step 4 Run the following command to configure environment variables:

source bigdata_env

Step 5 If multiple Storm instances are installed, run the following command to load the environment variables of a specific instance when running the Storm command to submit the topology. Otherwise, skip this step. The following command uses the instance Storm-2 as an example.

source Storm-2/component_env

Step 6 For clusters with Kerberos authentication enabled, run the following command to authenticate the user. For clusters with Kerberos authentication disabled, skip this step.

kinit *Storm user*

Step 7 For versions earlier than MRS 3.x, run the following command to submit the Storm topology:

storm jar *Path of the topology package Class name of the topology Main method Topology name*

If the following information is displayed, the topology is submitted successfully.

```
Finished submitting topology: topo1
```

NOTE

- To support sampling messages, add the **topology.debug** and **topology.eventlogger.executors** parameters.
- Data processing methods vary with topologies. The topology in the example generates characters randomly and separates character strings. To query the processing status, enable the sampling function and perform operations according to **Querying Storm Topology Logs**.

Step 8 Run the following command to submit a topology task for MRS 3.x or later:

storm jar *topology-jar-path class input parameter list*

- *topology-jar-path* indicates the path of the JAR file of the topology.
- *class* indicates the class name of the main method used by the topology.
- *Input parameter list* includes input parameters of the main method used by the topology.

If the following information is displayed, the topology is submitted successfully:

```
Finished submitting topology: topology1
```

NOTE

- The login authentication user must correspond to the loaded environment variable (**component_env**). Otherwise, an error occurs when you run the **storm** command to submit the topology task.
- After the client environment variable is loaded and the corresponding user login succeeds, the user can run the Storm command on any Storm client to submit the topology task. After the command is executed, the successfully submitted topology is still in the Storm cluster of the user.
- If cluster domain name is modified, you need to reset the domain name before submitting the topology. Run the cql statement.

Step 9 Run the following command to query Storm topologies. For clusters with Kerberos authentication enabled, only users in the **stormadmin** or **storm** group can query all topologies.

storm list

----End

12.25.4 Accessing the Storm Web UI

Scenario

The Storm web UI provides a graphical interface for using Storm.

The following information can be queried on the Storm web UI:

- Storm cluster summary
- Nimbus summary

- Topology summary
- Supervisor summary
- Nimbus configurations

Prerequisites

- The password of user **admin** has been obtained. The password of user **admin** is specified by you during the cluster creation.
- If a user other than **admin** is used to access the Storm web UI, the user must be added to the **storm** or **stormadmin** user group.

Procedure

Step 1 Access the component management page.

- For versions earlier than MRS 3.*x*, click the cluster name to go to the cluster details page and choose **Components**.

  NOTE

  If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)

- For MRS 3.*x* or later, log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Choose **Cluster** > *Name of the desired cluster* > **Services**.

Step 2 Log in to the Storm WebUI.

- For versions earlier than MRS 3.x: Choose **Storm**. On the **Storm Summary** area, click any UI link on the right side of **Storm Web UI** to open the Storm web UI.

  NOTE

  When accessing the Storm web UI for the first time, you must add the address to the trusted site list.

- For MRS 3.x or later, choose **Storm** > **Overview**. In the **Basic Information** area, click any UI link on the right side of **Storm Web UI** to open the Storm web UI.

----End

Related Tasks

- Click a topology name to view details, status, Spouts information, Bolts information, and configuration information of the topology.
- In the **Topology actions** area, click **Activate**, **Deactivate**, **Rebalance**, **Kill**, **Debug**, **Stop Debug**, and **Change Log Level** to activate, deactivate, redeploy, delete, debug, and stop debugging the topology, and modify the log levels, respectively. You need to set the waiting time for the redeployment and deletion operations. The unit is second.
- In the **Topology Visualization** area, click **Show Visualization** to visualize a topology. After the topology is visualized, the WebUI displays the topology structure.

12.25.5 Managing Storm Topologies

Scenario

You can manage Storm topologies on the Storm web UI. Users in the **storm** group can manage only the topology tasks submitted by themselves, while users in the **stormadmin** group can manage all topology tasks.

Procedure

Step 1 For details about how to access the Storm WebUI, see **Accessing the Storm Web UI**.

Step 2 In the **Topology summary** area, click the desired topology.

Step 3 Use options in **Topology actions** to manage the Storm topology.

- Activating a topology

  Click **Activate** to activate the topology.

- Deactivating a topology

  Click **Deactivate** to deactivate the topology.

- Re-deploying a topology

  Click **Rebalance** and specify the wait time (in seconds) of re-deployment. Generally, if the number of nodes in a cluster changes, the topology can be re-deployed to maximize resource usage.

- Deleting a topology

  Click **Kill** and specify the wait time (in seconds) of the deletion.

- Starting or stopping sampling messages

  Click **Debug**. In the dialog box displayed, specify the percentage of the sampled data volume. For example, if the value is set to **10**, 10% of data is sampled.

  To stop sampling, click **Stop Debug**.

  NOTE

  This function is available only if the sampling function is enabled when the topology is submitted. For details about querying data processing information, see **Querying Storm Topology Logs**.

- Modifying the topology log level

  Click **Change Log Level** to specify a new log level.

Step 4 Displaying a topology

In the **Topology Visualization** area, click **Show Visualization** to visualize the topology.

----End

12.25.6 Querying Storm Topology Logs

Scenario

You can query topology logs to check the execution of a Storm topology in a worker process. To query the data processing logs of a topology, enable the **Debug** function when submitting the topology. Only streaming clusters with Kerberos authentication enabled support this function. In addition, the user who queries topology logs must be the one who submits the topology or a member of the **stormadmin** group.

Prerequisites

- The network of the working environment has been configured.
- The sampling function has been enabled for the topology.

Querying Worker Process Logs

Step 1 For details about how to access the Storm WebUI, see **Accessing the Storm Web UI**.

Step 2 In the **Topology Summary** area, click the desired topology to view details.

Step 3 Click the desired **Spouts** or **Bolts** task. In the **Executors (All time)** area, click a port in **Port** to view detailed logs.

----End

Querying Data Processing Logs of a Topology

Step 1 For details about how to access the Storm WebUI, see **Accessing the Storm Web UI**.

Step 2 In the **Topology Summary** area, click the desired topology to view details.

Step 3 Click **Debug**, specify the data sampling ratio, and click **OK**.

Step 4 Click the **Spouts** or **Bolts** task. In **Component summary**, click **events** to view data processing logs.

----End

12.25.7 Storm Common Parameters

This section applies to MRS 3.*x* or later.

Navigation Path

For details about how to set parameters, see **Modifying Cluster Service Configuration Parameters**.

Parameter Description

Table 12-435 Parameter description

Parameter	Description	Default Value
supervisor.slots.ports	Specifies the list of ports that can run workers on the supervisor. Each worker occupies a port, and each port runs only one worker. This parameter is used to set the number of workers that can run on each server. Ports range from 1024 to 65535, and ports are separated by commas (,).	6700,6701,6702,6703
WORKER_GC_OPTS	Specifies the JVM option used for supervisor to start worker. It is recommended that you set this parameter based on memory usage of a service. For simple service processing, the recommended value is **-Xmx1G**. If window cache is used, the value of this parameter is calculated based on the following formula: Size of each record x Period x 2	-Xms1G -Xmx1G -XX:+UseG1GC -XX:+PrintGCDetails -Xloggc:artifacts/gc.log -XX:+PrintGCDateStamps -XX:+PrintGCTimeStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=artifacts/heapdump
default.schedule.mode	Specifies the default scheduling mode of the scheduler. Options are as follows: ● **AVERAGE**: indicates that the scheduling mechanism that uses the number of idle slots as the priority is used. ● **RATE**: indicates that the scheduling mechanism that uses the rate of idle slots as the priority is used.	AVERAGE
nimbus.thrift.threads	Set the maximum number of connection threads when the active Nimbus externally provides services. If the Storm cluster is large and the number of Supervisor instances is large, increase connection threads.	512

12.25.8 Configuring a Storm Service User Password Policy

Scenario

This section applies to MRS 3.*x* or later.

After submitting a topology task, a Storm service user must ensure that the task continuously runs. During topology running, the worker process may need to restart to ensure continuous topology work. If the password of a service user is changed or the number of days that a password is used exceeds the maximum number specified in a password policy, topology running may be affected. A system administrator must configure a separate password policy for Storm service users based on enterprise security requirements.

NOTE

If a separate password policy is not configured for Storm service users, an old topology can be deleted and then submitted again after a service user password is changed so that the topology can continuous run.

Impact on the System

- After a separate password policy is configured for a Storm service user, the user is not affected by **Password Policy** on the Manager page.
- If a separate password policy is configured for a Storm service user and cross-cluster entrusted relationships are configured, a password must be reset for the Storm service user on Manager based on the password policy.

Prerequisites

A system administrator has understood service requirements and created a **Human-Machine** user, for example, **testpol**.

Procedure

Step 1 Log in to any node in the cluster as user **omm**.

Step 2 Run the following command to disable logout upon timeout:

TMOUT=0

NOTE

After the operations in this section are complete, run the **TMOUT=***Timeout interval* command to restore the timeout interval in a timely manner. For example, **TMOUT=600** indicates that a user is logged out if the user does not perform any operation within 600 seconds.

Step 3 Run the following commands to export the environment variables:

EXECUTABLE_HOME="${CONTROLLER_HOME}/kerberos_user_specific_binay/kerberos"

LD_LIBRARY_PATH=${EXECUTABLE_HOME}/lib:$LD_LIBRARY_PATH

PATH=${EXECUTABLE_HOME}/bin:$PATH

Step 4 Run the following command and enter the Kerberos administrator password to log in to the Kerberos console:

kadmin -p kadmin/admin

NOTE

For initial use, the **kadmin/admin** password must be changed for the **kadmin/admin** user.

If the following information is displayed, you have successfully logged in to the Kerberos console.

```
kadmin:
```

Step 5 Run the following command to check details about the created **Human-Machine** user:

getprinc*Username*

Sample command for viewing details about the **testpol** user:

getprinc testpol

If the following information is displayed, the specified user has used the default password policy:

```
Principal: testpol@<System domain name>
......
Policy: default
```

Step 6 Run the following command to create a separate password policy, such as **streampol**, for the Storm service user:

addpol -maxlife 0day -minlife 0sec -history 1 -maxfailure 5 -failurecountinterval 5min -lockoutduration 5min -minlength 8 -minclasses 4 streampol

In the command, **-maxlife** indicates the maximum validity period of a password, and **0day** indicates that a password will never expire.

Step 7 Run the following command to view the newly created policy **streampol**:

getpol streampol

If the following information is displayed, the new policy specifies that the password will never expire:

```
Policy: streampol
 Maximum password life: 0 days 00:00:00
......
```

Step 8 Run the following command to apply the new policy **streampol** to the **testpol** Storm user:

modprinc -policy streampol testpol

In the command, **streampol** indicates a policy name, and **testpol** indicates a username.

If the following information is displayed, the properties of the specified user have been modified:

```
Principal "testpol@<System domain name>" modified.
```

Step 9 Run the following command to view current information about the **testpol** Storm user:

getprinc testpol

If the following information is displayed, the specified user has used the new password policy:

```
Principal: testpol@<System domain name>
.....
 Policy: streampol
```

----End

12.25.9 Migrating Storm Services to Flink

12.25.9.1 Overview

This section applies to MRS 3.*x* or later.

From 0.10.0, Flink provides a set of APIs to smoothly migrate services compiled using Storm APIs to the Flink platform. This can be used in most of the service scenarios.

Flink supports the following service migration modes:

1. Complete migration of Storm services: Convert and run a complete Storm topology developed by Storm APIs.
2. Embedded migration of Storm services: Storm code is embedded in DataStream of Flink, for example, Spout/Bolt compiled using Storm APIs.

Flink provides the flink-storm package for the preceding service migration.

12.25.9.2 Completely Migrating Storm Services

Scenarios

This section describes how to convert and run a complete Storm topology developed using Storm API.

Procedure

Step 1 Open the Storm service project, modify the POM file of the project, and add the reference of **flink-storm_2.11**, **flink-core**, and **flink-streaming-java_2.11**. The following figure shows an example.

```
<dependency>
   <groupId>org.apache.flink</groupId>
   <artifactId>flink-storm_2.11</artifactId>
   <version>1.4.0</version>
   <exclusions>
      <exclusion>
         <groupId>*</groupId>
         <artifactId>*</artifactId>
      </exclusion>
   </exclusions>
</dependency>
```

```
<dependency>
   <groupId>org.apache.flink</groupId>
   <artifactId>flink-core</artifactId>
   <version>1.4.0</version>
   <exclusions>
      <exclusion>
         <groupId>*</groupId>
         <artifactId>*</artifactId>
      </exclusion>
   </exclusions>
</dependency>
```

```
<dependency>
   <groupId>org.apache.flink</groupId>
   <artifactId>flink-streaming-java_2.11</artifactId>
   <version>1.4.0</version>
   <exclusions>
      <exclusion>
         <groupId>*</groupId>
         <artifactId>*</artifactId>
      </exclusion>
   </exclusions>
</dependency>
```

NOTE

If the project is not a non-Maven project, manually collect the preceding JAR packages and add them to the *classpath* environment variable of the project.

Step 2 Modify the code for submission of the topology. The following uses WordCount as an example:

1. Keep the structure of the Storm topology unchanged, including the Spout and Bolt developed using Storm API.

```
TopologyBuilder builder = new TopologyBuilder();
builder.setSpout("spout", new RandomSentenceSpout(), 5);
builder.setBolt("split", new SplitSentenceBolt(), 8).shuffleGrouping("spout");
builder.setBolt("count", new WordCountBolt(), 12).fieldsGrouping("split", new Fields("word"));
```

2. Modify the code for submission of the topology. An example is described as follows:

```
Config conf = new Config();
conf.setNumWorkers(3);
StormSubmitter.submitTopology("word-count", conf, builder.createTopology());
```

Perform the following operations:

```
Config conf = new Config();
 conf.setNumWorkers(3);
 //converts Storm Config to StormConfig of Flink.
 StormConfig stormConfig = new StormConfig(conf);
 //Construct FlinkTopology using TopologBuilder of Storm.
 FlinkTopology topology = FlinkTopology.createTopology(builder);
 //Obtain the Stream execution environment.
 StreamExecutionEnvironment env = topology.getExecutionEnvironment();
 //Set StormConfig to the environment variable of Job to construct Bolt and Spout.
 //If StormConfig is not required during the initialization of Bolt and Spout, you do not need to set this
parameter.
 env.getConfig().setGlobalJobParameters(stormConfig);
 //Submit the topology.
 topology.execute();
```

3. After the package is repacked, run the following command to submit the package:

   flink run -class {MainClass} WordCount.jar

----End

12.25.9.3 Performing Embedded Service Migration

Scenarios

This section describes how to embed Storm code in DataStream of Flink in embedded migration mode. For example, the code of Spout or Bolt compiled using Storm API is embedded.

Procedure

Step 1 In Flink, perform embedded conversion to Spout and Bolt in the Storm topology to convert them to Flink operators. The following is an example of the code:

```
//set up the execution environment
final StreamExecutionEnvironment env = StreamExecutionEnvironment.getExecutionEnvironment();
//get input data
final DataStream<String> text = getTextDataStream(env);
final DataStream<Tuple2<String, Integer>> counts = text
  //split up the lines in pairs (2-tuples) containing: (word,1)
  //this is done by a bolt that is wrapped accordingly
  .transform("CountBolt",
    TypeExtractor.getForObject(new Tuple2<String, Integer>("", 0)),
    new BoltWrapper<String, Tuple2<String, Integer>>(new CountBolt()))
  //group by the tuple field "0" and sum up tuple field "1"
  .keyBy(0).sum(1);
// execute program
env.execute("Streaming WordCount with bolt tokenizer");
```

Step 2 After the modification, run the following command to submit the modification:

flink run -class {MainClass} WordCount.jar

----End

12.25.9.4 Migrating Services of External Security Components Interconnected with Storm

Migrating Services for Interconnecting Storm with HDFS and HBase

If the Storm services use the **storm-hdfs** or **storm-hbase** plug-in package for interconnection, you need to specify the following security parameters when migrating Storm services as instructed in **Completely Migrating Storm Services**.

```
//Initialize Storm Config.
Config conf = new Config();

//Initialize the security plug-in list.
List<String> auto_tgts = new ArrayList<String>();
//Add the AutoTGT plug-in.
auto_tgts.add("org.apache.storm.security.auth.kerberos.AutoTGT");
//Add the AutoHDFS plug-in.
//If HBase is interconnected, use auto_tgts.add("org.apache.storm.hbase.security.AutoHBase") to replace the following:
auto_tgts.add("org.apache.storm.hdfs.common.security.AutoHDFS");

//Set security parameters.
conf.put(Config.TOPOLOGY_AUTO_CREDENTIALS, auto_tgts);
//Set the number of workers.
conf.setNumWorkers(3);

//Convert Storm Config to StormConfig of Flink.
StormConfig stormConfig = new StormConfig(conf);
```

```
//Construct FlinkTopology using TopologBuilder of Storm.
FlinkTopology topology = FlinkTopology.createTopology(builder);

//Obtain the StreamExecutionEnvironment.
StreamExecutionEnvironment env = topology.getExecutionEnvironment();

//Add StormConfig to the environment variable of Job to construct Bolt and Spout.
//If Config is not required during the initialization of Bolt and Spout, do not set this parameter.
env.getConfig().setGlobalJobParameters(stormConfig);

//Submit the topology.
topology.execute();
```

After the preceding security plug-in is configured, unnecessary logins during the initialization of HDFSBolt and HBaseBolt are avoided because the security context has been configured in Flink.

Migrating Services of Storm Interconnected with Other Security Components

If the plug-in packages, such as **storm-kakfa-client** and **storm-solr** are used for interconnection between Storm and other components for service migration, the previously configured security plug-ins need to be deleted.

```
List<String> auto_tgts = new ArrayList<String>();
//keytab mode
auto_tgts.add("org.apache.storm.security.auth.kerberos.AutoTGTFromKeytab");

//Write the plug-in list configured on the client to the specified config parameter.
//Mandatory in security mode
//This configuration is not required in common mode, and you can comment out the following line.
conf.put(Config.TOPOLOGY_AUTO_CREDENTIALS, auto_tgts);
```

The AutoTGTFromKeytab plug-in must be deleted during service migration. Otherwise, the login will fail when Bolt or Spout is initialized.

12.25.10 Storm Log Introduction

This section applies to MRS 3.*x* or later.

Log Description

Log paths: The default paths of Storm log files are **/var/log/Bigdata/storm/***Role name* (run logs) and **/var/log/Bigdata/audit/storm/***Role name* (audit logs).

- Nimbus: **/var/log/Bigdata/storm/nimbus** (run logs) and **/var/log/Bigdata/audit/storm/nimbus** (audit logs)
- Supervisor: **/var/log/Bigdata/storm/supervisor** (run logs) and **/var/log/Bigdata/audit/storm/supervisor** (audit logs)
- UI: **/var/log/Bigdata/storm/ui** (run logs) and **/var/log/Bigdata/audit/storm/ui** (audit logs)
- Logviewer: **/var/log/Bigdata/storm/logviewer** (run logs) and **/var/log/Bigdata/audit/storm/logviewer** (audit logs)

Log archive rule: The automatic Storm log compression function is enabled. By default, when the size of logs exceeds 10 MB, logs are automatically compressed into a log file named in the following format: *<Original log name>***.log.***[ID]***.gz**. A

maximum of 20 latest compressed files are reserved by default. You can configure the number of compressed files and the compression threshold.

Names of compressed audit log files are in the format of **audit.log.***[yyyy-MM-dd]*.*[ID]*.**zip**. These files permanently exist.

Table 12-436 Storm log list

Log Type	Log File Name	Description
Run log	nimbus/access.log	Nimbus user access log
	nimbus/nimbus-<PID>-gc.log	GC log of the Nimbus process
	nimbus/checkavailable.log	Nimbus availability check log
	nimbus/checkService.log	Nimbus serviceability check log
	nimbus/metrics.log	Nimbus monitoring statistics log
	nimbus/nimbus.log	Run log of the Nimbus process
	nimbus/postinstall.log	Work log after Nimbus installation
	nimbus/prestart.log	Work log before Nimbus startup
	nimbus/start.log	Work log of Nimbus startup
	nimbus/stop.log	Work log of Nimbus shutdown
	supervisor/access.log	Supervisor access log
	supervisor/metrics.log	Supervisor monitoring statistics log
	supervisor/postinstall.log	Work log after supervisor installation
	supervisor/prestart.log	Work log before supervisor startup
	supervisor/start.log	Work log of supervisor startup
	supervisor/stop.log	Work log of supervisor shutdown
	supervisor/supervisor.log	Run log of the supervisor process

Log Type	Log File Name	Description
	supervisor/supervisor-<PID>-gc.log	GC log of the supervisor process
	ui/access.log	UI access log
	ui/metric.log	UI monitoring statistics log
	ui/ui-<PID>-gc.log	GC log of the UI process
	ui/postinstall.log	Work log after UI installation
	ui/prestart.log	Work log before UI startup
	ui/start.log	Work log of UI startup
	ui/stop.log	Work log of UI shutdown
	ui/ui.log	Run log of the UI process
	logviewer/access.log	Logviewer access log
	logviewer/metric.log	Logviewer monitoring statistics log
	logviewer/logviewer-<PID>-gc.log	GC log file of the logviewer process
	logviewer/logviewer.log	Run log of the logviewer process
	logviewer/postinstall.log	Work log after logviewer installation
	logviewer/prestart.log	Work log before logviewer startup
	logviewer/start.log	Work log of logviewer startup
	logviewer/stop.log	Work log of logviewer shutdown
	supervisor/[topologyId]-worker-[*Port number*].log	Run log of the Worker process. One port occupies one log file. By default, the system contains five ports: 29100, 29101, 29102, 29103 and 29304.

Log Type	Log File Name	Description
	supervisor/metadata/[topologyid]-worker-[*Port number*].yaml	Worker log metadata file, which is used by logviewer to delete logs. This file is automatically deleted by the logviewer log deletion thread based on certain conditions.
	nimbus/cleanup.log	Cleanup log of Nimbus uninstallation
	logviewer/cleanup.log	Cleanup log of logviewer uninstallation
	ui/cleanup.log	Cleanup log of UI uninstallation
	supervisor/cleanup.log	Cleanup log of supervisor uninstallation
	leader_switch.log	Run log file that records the Storm active/standby switchover
Audit log	nimbus/audit.log	Nimbus audit log
	ui/audit.log	UI audit log
	supervisor/audit.log	Supervisor audit log
	logviewer/audit	Logviewer audit log

Log Levels

Table 12-437 describes the log levels supported by Storm.

Levels of run logs and audit logs are ERROR, WARN, INFO, and DEBUG from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-437 Log levels

Level	Description
ERROR	Logs of this level record error information about system running.
WARN	Logs of this level record exception information about the current event processing.

Level	Description
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of Storm by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Save the configuration. In the displayed dialog box, click **OK** to make the configurations take effect.

----End

Log Format

The following table lists the Storm log formats:

Table 12-438 Log Formats

Log Type	Format	Example
Run log	%d{yyyy-MM-dd HH:mm:ss,SSS} | %-5p | [%t] | %m | %logger (%F:%L) %n	2015-03-11 23:04:00,241 | INFO | [RMI TCP Connection(2646)-10.0.0.2] | The baseSleepTimeMs [1000] the maxSleepTimeMs [1000] the maxRetries [1] | backtype.storm.utils.StormBoundedExponentialBackoffRetry (StormBoundedExponentialBackoffRetry.java:46)
	<yyyy-MM-dd HH:mm:ss,SSS><HostName><RoleName><logLevel><Message>	2017-03-28 02:57:52 493 10-5-146-1 storm- INFO Nimbus start normally

Log Type	Format	Example
Audit log	*<Username><User IP address><Time><Operation><Operation object><Operation result>*	UserName=storm/hadoop, UserIP=10.10.0.2, Time=Tue Mar 10 01:15:35 CST 2015, Operation=Kill, Resource=test, Result=Success

12.25.11 Performance Tuning

12.25.11.1 Storm Performance Tuning

Scenario

You can modify Storm parameters to improve Storm performance in specific service scenarios.

This section applies to MRS 3.x or later.

Modify the service configuration parameters. For details, see **Modifying Cluster Service Configuration Parameters**.

Topology Tuning

This task enables you to optimize topologies to improve efficiency for Storm to process data. It is recommended that topologies be optimized in scenarios with lower reliability requirements.

Table 12-439 Tuning parameters

Parameter	Default Value	Scenario
topology.acker.executors	null	Specifies the number of acker executors. If a service application has lower reliability requirements and certain data does not need to be processed, this parameter can be set to **null** or **0** so that you can set acker off, flow control is weakened, and message delay is not calculated. This improves performance.

Parameter	Default Value	Scenario
topology.max.spout.pending	null	Specifies the number of messages cached by spout. The parameter value takes effect only when acker is not **0** or **null**. Spout adds each message sent to downstream bolt into the pending queue. The message is removed from the queue after downstream bolt processes the message and the processing is confirmed. When the pending queue is full, spout stops sending messages. Increasing the pending value improves the message throughput of spout per second but prolongs the delay.
topology.transfer.buffer.size	32	Specifies the size of the Distuptor message queue for each worker process. It is recommended that the size be between 4 to 32. Increasing the queue size improves the throughput but may prolong the delay.
RES_CPUSET_PERCENTAGE	80	Specifies the percentage of physical CPU resources used by the supervisor role instance (including startup and management worker processes) on each node. Adjust the parameter value based on service volume requirements of the node on which the supervisor exists, to optimize CPU usage.

JVM Tuning

If an application must occupy more memory resources to process a large volume of data and the size of worker memory is greater than 2 GB, the G1 garbage collection algorithm is recommended.

Table 12-440 Tuning parameters

Parameter	Default Value	Scenario
WORKER_GC_OPTS	-Xms1G -Xmx1G -XX:+UseG1GC -XX:+PrintGCDetails -Xloggc:artifacts/gc.log -XX:+PrintGCDateStamps -XX:+PrintGCTimeStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=artifacts/heapdump	If an application must occupy more memory resources to process a large volume of data and the size of worker memory is greater than 2 GB, the G1 garbage collection algorithm is recommended. In this case, change the parameter value to **-Xms2G -Xmx5G -XX:+UseG1GC**.

12.26 Using Tez

12.26.1 Precautions

This section applies to MRS 3.*x* or later.

12.26.2 Common Tez Parameters

Navigation path for setting parameters:

On Manager, choose **Cluster** > **Service** > **Tez** > **Configuration** > **All Configurations**. Enter a parameter name in the search box.

Parameter description

Table 12-441 Parameter description

Parameter	Description	Default Value
property.tez.log.dir	TezUI log directory	/var/log/Bigdata/tez/tezui
property.tez.log.level	TezUI log level	INFO

12.26.3 Accessing TezUI

Tez displays the Tez task execution process on a GUI. You can view the task execution details on the GUI.

Prerequisite

The TimelineServer instance of the Yarn service has been installed.

How to Use

Log in to Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. On Manager, choose **Cluster** > **Services** > **Tez**. Click the link on the right of **Tez WebUI** in the **Basic Information** area, and go to Tez web UI. You can view the details about Tez task execution.

12.26.4 Log Overview

Log Description

Log path: The default save path of Tez logs is **/var/log/Bigdata/tez/***role name*.

TezUI: **/var/log/Bigdata/tez/tezui** (run logs) and **/var/log/Bigdata/audit/tez/tezui** (audit logs)

Log archive rule: The automatic compression and archiving function of Tez is enabled. By default, when the size of a log file exceeds 20 MB (which is adjustable), the log file is automatically compressed. The naming rule of the compressed log file is as follows: <*Original log file name*>-<*yyyy-mm-dd_hh-mm-ss*>.[*ID*].**log.zip** A maximum of 20 latest compressed files are retained. The number of compressed files and compression threshold can be configured.

Table 12-442 Tez log list

Log Type	Name	Description
Run log	tezui.out	Log file that records TezUI running environment information

Log Type	Name	Description
	tezui.log	Run log of the TezUI process
	tezui-omm-<*Date*>-gc.log.<*No.*>	GC log of the TezUI process
	prestartDetail.log	Work logs generated before the TezUI is started
	check-serviceDetail.log	Log file that records whether the TezUI service starts successfully
	postinstallDetail.log	Work logs after the TezUI is installed
	startDetail.log	Startup log of the TezUI process
	stopDetail.log	Stop log of the TezUI process
Audit log	tezui-audit.log	TezUI audit log

Log Level

Table 12-443 describes the log levels supported by TezUI.

Levels of run logs are ERROR, WARN, INFO, and DEBUG from the highest to the lowest priority. Run logs of equal or higher levels are recorded. The higher the specified log level, the fewer the logs recorded.

Table 12-443 Log levels

Level	Description
ERROR	Logs of this level record error information about system running.
WARN	Exception information about the current event processing
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Log in to Manager.

Step 2 Choose **Cluster** > **Service** > **Tez** > **Configuration**.

Step 3 Select **All Configurations**.

Step 4 In the navigation pane, choose **TezUI** > **Log**.

Step 5 Select a desired log level.

Step 6 Click **Save**. In the dialog box that is displayed, click **OK** to save the configuration.

Step 7 Click **Instance**, select the **TezUI** role, choose **More** > **Restart Instance**, enter the user password, and click **OK** in the dialog box that is displayed.

Step 8 Wait until the instance is restarted for the configuration to take effect.

----End

Log Format

The following table lists the Tez log formats.

Table 12-444 Log formats

Log Type	Format	Example
Run log	<yyyy-MM-dd HH:mm:ss,SSS>| <LogLevel>|<Thread that generates the log>| <Message in the log>| <Location of the log event>	2020-07-31 11:44:21,378 | INFO | TezUI-health-check | Start health check | com.XXX.tez.HealthCheck.run(HealthCheck.java:30)
Audit logs	<yyyy-MM-dd HH:mm:ss,SSS>| <LogLevel>|<Thread that generates the log>|<User Name><User IP><Time><Operation><Resource><Result><Detail >|< Location of the log event >	2018-12-24 12:16:25,319 | INFO | HiveServer2-Handler-Pool: Thread-185 | UserName=hive UserIP=10.153.2.204 Time=2018/12/24 12:16:25 Operation=CloseSession Result=SUCCESS Detail= | org.apache.hive.service.cli.thrift.ThriftCLIService.logAuditEvent(ThriftCLIService.java:434)

12.26.5 Common Issues

12.26.5.1 TezUI Cannot Display Tez Task Execution Details

Question

After a user logs in to Manager and switches to the Tez web UI, the submitted Tez tasks are not displayed.

Answer

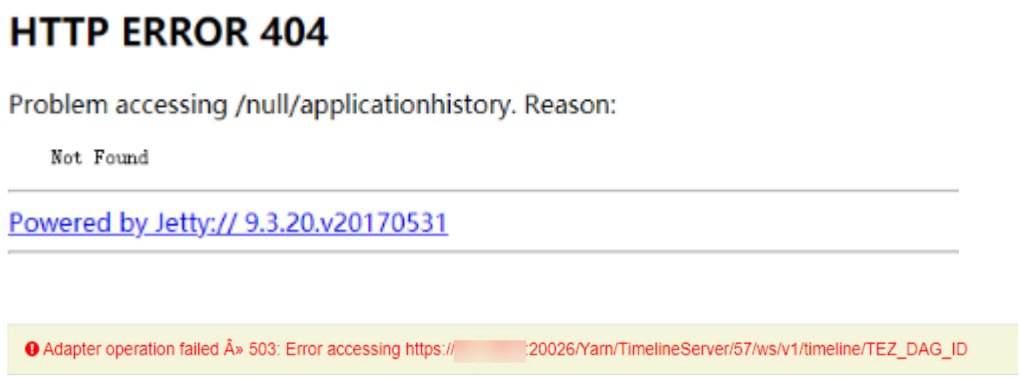
The Tez task data displayed on the Tez WebUI requires the support of TimelineServer of Yarn. Ensure that TimelineServer has been enabled and is running properly before the task is submitted.

When setting the Hive execution engine to Tez, you need to set **yarn.timeline-service.enabled** to **true**. For details, see **Switching the Hive Execution Engine to Tez**.

12.26.5.2 Error Occurs When a User Switches to the Tez Web UI

Question

When a user logs in to Manager and switches to the Tez web UI, error 404 or 503 is displayed.



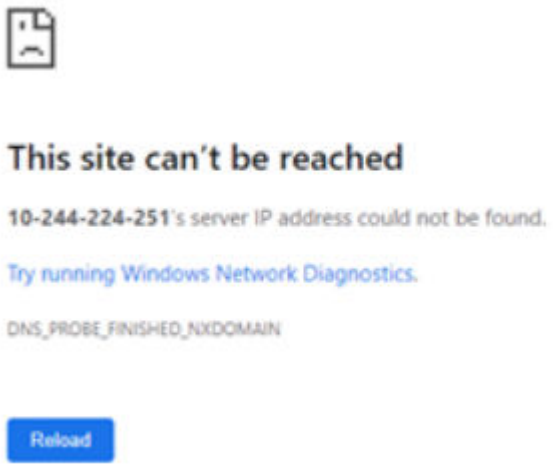
Answer

The Tez web UI depends on the TimelineServer instance of Yarn. Therefore, TimelineServer must be installed in advance and in the **Good** state.

12.26.5.3 Yarn Logs Cannot Be Viewed on the TezUI Page

Question

A user logs in to the Tez web UI and clicks **Logs**, but the Yarn log page fails to be displayed and data cannot be loaded.



Answer

Currently, the hostname is used for the access to the Yarn log page from the Tez web UI. Therefore, you need to configure the mapping between the hostname and IP address on the Windows host. Perform the following steps:

Modify the **C:\Windows\System32\drivers\etc\hosts** file on the Windows host and add a line indicating the mapping between the host name and IP address, for example, **10.244.224.45 10-044-224-45**. Save the modification and access the host again.

12.26.5.4 Table Data Is Empty on the TezUI HiveQueries Page

Question

A user logs in to Manager and switches to the Tez web UI page, but no data for the submitted task is displayed on the **Hive Queries** page.

Answer

To display task data on the **Hive Queries** page on the Tez web UI, you need to set the following parameters:

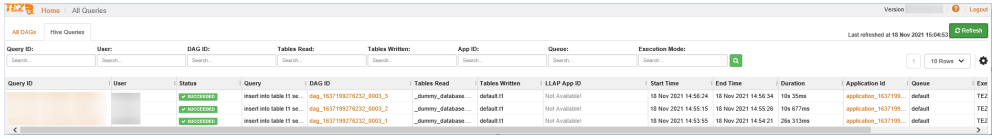
On FusionInsight Manager, choose **Cluster** > **Service** > **Hive** and click the **Configurations** tab and then **All Configurations**. In the navigation pane on the left, choose **HiveServer** > **Customization**. Add the following configuration to **hive-site.xml**:

Attribute	Attribute Value
hive.exec.pre.hooks	org.apache.hadoop.hive.ql.hooks.ATSHook
hive.exec.post.hooks	org.apache.hadoop.hive.ql.hooks.ATSHook
hive.exec.failure.hooks	org.apache.hadoop.hive.ql.hooks.ATSHook

NOTE

Data display on TezUI depends on the TimelineServer instance of Yarn. If the TimelineServer instance is faulty or not started, you need to set **yarn.timeline-service.enabled** to **false** in **yarn-site.xml**. Otherwise, the Hive task fails to be executed.

After you configure the parameters and re-execute the Hive task, data can be displayed on the **Hive Queries** page. However, data of previous tasks cannot be displayed.



12.27 Using Yarn

12.27.1 Common YARN Parameters

Allocating Queue Resources

The Yarn service provides queues for users. Users allocate system resources to each queue. After the configuration is complete, you can click **Refresh Queue** or restart the Yarn service for the configuration to take effect.

Navigation path for setting parameters:

For versions earlier than MRS 3.x, perform the following operations:

On the MRS console, choose **Tenants** > **Resource Distribution Policies**.

The following uses the **default** queue as an example. The configurations of other queues are similar. Click **Modify** to edit the parameters.

Table 12-445 Parameter description

Parameter	Description	Default Value
Capacity	Queue resource capacity (percentage). Ensure that the capacity requirement of each queue is satisfied when the system is busy. If only a few application programs are running in a queue, the remaining resource of the queue can be shared with other queues. Note that the total capacity of all queues must be smaller than 100.	20
Maximum Capacity	Maximum queue resource usage (percentage). Due to resource sharing, the resources used by a queue may exceed its capacity. The maximum resource usage can be limited using this parameter.	100

For MRS 3.*x* or later, perform the following operations:

On Manager, choose **Tenant Resources** > **Dynamic Resource Plan** > **Queue Configuration**.

The following uses the **default** tenant who modifies the Superior scheduler as an example. The configurations of other queues are similar. Click **Modify** to edit the parameters.

Table 12-446 Queue configuration parameters

Parameter	Description
Max Master Shares(%)	Indicates the maximum percentage of resources occupied by all ApplicationMasters in the current queue.
Max Allocated vCores	Indicates the maximum number of cores that can be allocated to a single YARN container in the current queue. The default value is **-1**, indicating that the number of cores is not limited within the value range.
Max Allocated Memory(MB)	Indicates the maximum memory that can be allocated to a single YARN container in the current queue. The default value is **-1**, indicating that the memory is not limited within the value range.
Max Running Apps	Maximum number of tasks that can be executed at the same time in the current queue. The default value is **-1**, indicating that the number is not limited within the value range (the meaning is the same if the value is empty). The value 0 indicates that the task cannot be executed. The value ranges from -1 to 2147483647.
Max Running Apps per User	Maximum number of tasks that can be executed by each user in the current queue at the same time. The default value is **-1**, indicating that the number is not limited within the value range. If the value is **0**, the task cannot be executed. The value ranges from -1 to 2147483647.
Max Pending Apps	Maximum number of tasks that can be suspended at the same time in the current queue. The default value is **-1**, indicating that the number is not limited within the value range (the meaning is the same if the value is empty). The value **0** indicates that tasks cannot be suspended. The value ranges from -1 to 2147483647.
Resource Allocation Rule	Indicates the rule for allocating resources to different tasks of a user. The rule can be FIFO or FAIR. If a user submits multiple tasks in the current queue and the rule is FIFO, the tasks are executed one by one in sequential order; if the rule is FAIR, resources are evenly allocated to all tasks.

Parameter	Description
Default Resource Label	Indicates that tasks are executed on a node with a specified resource label.
Active	• **ACTIVE**: indicates that the current queue can receive and execute tasks. • **INACTIVE**: indicates that the current queue can receive but cannot execute tasks. Tasks submitted to the queue are suspended.
Open	• **OPEN**: indicates that the current queue is opened. • **CLOSED**: indicates that the current queue is closed. Tasks submitted to the queue are rejected.

Displaying Container Logs on the Web UI

By default, the system collects container logs to HDFS. If you do not need to collect container logs to HDFS, configure the parameters in **Table 12-447**. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-447 Parameter description

Parameter	Description	Default Value
yarn.log-aggregation-enable	Select whether to collect container logs to HDFS. • If the parameter is set to **true**, container logs are collected to an HDFS directory. The default directory is **{yarn.nodemanager.remote-app-log-dir}/${user}/{thisParam}**. You can set the directory by setting the **yarn.nodemanager.remote-app-log-dir-suffix** parameter on the web UI. • If this parameter is set to **false**, container logs will not be collected to HDFS. After changing the parameter value, restart the Yarn service for the setting to take effect. **NOTE** The container logs that are generated before the parameter is set to **false** and the setting takes effect cannot be obtained from the web UI. You can obtain container logs from the directory specified by the **yarn.nodemanager.remote-app-log-dir-suffix** parameter before the setting takes effect. If you want to view the logs generated before on the web UI, you are advised to set this parameter to **true**.	true

Increasing the Number of Historical Jobs to Be Displayed on the web UI

By default, the Yarn web UI supports task list pagination. A maximum of 5,000 historical jobs can be displayed on each page, and a maximum of 10,000 historical jobs can be retained. If you need to view more jobs on the WebUI, configure parameters by referring to **Table 12-448**. For details, see **Modifying Cluster Service Configuration Parameters**.

Table 12-448 Parameter description

Parameter	Description	Default Value
yarn.resourcemanager.max-completed-applications	Set the total number of historical jobs to be displayed on the web UI.	10000
yarn.resourcemanager.webapp.pagination.enable	Select whether to enable the job list background pagination function for the Yarn web UI.	true

Parameter	Description	Default Value
yarn.resourcemanager.webapp.pagination.threshold	Set the maximum number of jobs displayed on each page after the job list background pagination function of the Yarn web UI is enabled.	5000

NOTE

- If a large number of historical jobs are displayed, the performance will be affected and the time for opening the Yarn web UI will be increased. Therefore, you are advised to enable the background pagination function and modify the **yarn.resourcemanager.max-completed-applications** parameter according to the actual hardware performance.
- After changing the parameter value, restart the Yarn service for the setting to take effect.

12.27.2 Creating Yarn Roles

Scenario

This section describes how to create and configure a Yarn role. The Yarn role can be assigned with Yarn administrator permission and manage Yarn queue resources.

NOTE

If the current component uses Ranger for permission control, you need to configure permission management policies based on Ranger. Refer to **Adding a Ranger Access Permission Policy for Yarn** for clusters of MRS 3.*x* or later.

Prerequisites

- The system administrator has understood the service requirements.
- You have logged in to Manager.

Procedure

For versions earlier than MRS 3.x, perform the following operations:

Step 1 Choose **System** > **Manage Role** > **Create Role**.

Step 2 Click **Create Role** and fill in **Role Name** and **Description**.

Step 3 Set permissions. For details, see **Table 12-449**.

Yarn permissions:

- **Cluster Admin Operations**: Yarn administrator permissions.
- **Scheduler Queue**: queue resources management .

Table 12-449 Setting a role

Task	Operation
Setting the Yarn administrator permission	In the **Permission** table, click **Yarn** and select **Cluster Admin Operations**. **NOTE** The Yarn service needs to be restarted to set the Yarn administrator permission so that the saved role configuration can take effect.
Setting the permission for a user to submit tasks in a specified Yarn queue	1. In the **Permission** table, choose **Yarn** > **Scheduler Queue**. 2. In the **Permission** column of the specified queue, select **Submit**.
Setting the permission for a user to manage tasks in a specified Yarn queue	1. In the **Permission** table, choose **Yarn** > **Scheduler Queue**. 2. In the **Permission** column of the specified queue, select **Admin**.

If the Yarn role contains the **Submit** or **Manage** permission of a parent queue, the sub-queue inherits the permission by default, that is, the **Submit** or **Manage** permission is automatically added for the sub-queue. Permissions inherited by sub-queues will not be displayed as selected in the **Configure Resource Permission** table.

If you select only the **Submit** permission of a parent queue when setting the Yarn role, you need to manually specify the queue name when submitting tasks as a user with the permission of this role. Otherwise, when the parent queue has multiple sub-queues, the system does not automatically determine the queue to which the task is submitted and therefore submits the task to the **default** queue.

Step 4 Click **OK**.

----End

For MRS 3.*x* or later, perform the following operations:

Step 1 Choose System > Permission > Role.

Step 2 Click **Create Role** and set a role name and enter description.

Step 3 Refer **Table 12-450** to configure resource permissions for roles.

Yarn permissions:

- Cluster management: Yarn administrator permissions.
- Queue scheduling: queue resource management.

Table 12-450 Setting a role

Task	Operation
Setting the Yarn administrator permission	In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Yarn** > **Cluster Management**. **NOTE** The Yarn service needs to be restarted to set the Yarn administrator permission so that the saved role configuration can take effect.
Setting the permission for a user to submit tasks in a specified Yarn queue	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Yarn** > **Scheduling Queue** > **root**. 2. In the **Permission** column of the specified queue, select **Submit**.
Setting the permission for a user to manage tasks in a specified Yarn queue	1. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Yarn** > **Scheduling Queue** > **root**. 2. In the **Permission** column of the specified queue, select **Manage**.

If the Yarn role contains the **Submit** or **Manage** permission of a parent queue, the sub-queue inherits the permission by default, that is, the **Submit** or **Manage** permission is automatically added for the sub-queue. Permissions inherited by sub-queues will not be displayed as selected in the **Configure Resource Permission** table.

If you select only the **Submit** permission of a parent queue when setting the Yarn role, you need to manually specify the queue name when submitting tasks as a user with the permission of this role. Otherwise, when the parent queue has multiple sub-queues, the system does not automatically determine the queue to which the task is submitted and therefore submits the task to the **default** queue.

Step 4 Click **OK**.

----End

12.27.3 Using the YARN Client

Scenario

This section guides users to use a Yarn client in an O&M or service scenario.

Prerequisites

- The client has been installed.

  For example, the installation directory is **/opt/hadoopclient**. The client directory in the following operations is only an example. Change it to the actual installation directory.

- Service component users are created by the administrator as required. In security mode, machine-machine users need to download the keytab file. A human-machine user must change the password upon the first login. In common mode, you do not need to download the keytab file or change the password.

Using the Yarn Client

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/hadoopclient

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster is in security mode, run the following command to authenticate the user. In normal mode, user authentication is not required.

kinit *Component service user*

Step 5 Run the Yarn command. The following provides an example:

yarn application -list

----End

Client-related FAQs

1. What Do I Do When the Yarn Client Exits Abnormally and Error Message "java.lang.OutOfMemoryError" Is Displayed After the Yarn Client Command Is Run?

   This problem occurs because the memory required for running the Yarn client exceeds the upper limit (128 MB by default) set on the Yarn client. For clusters of MRS 3.x or later: You can modify **CLIENT_GC_OPTS** in *<Client installation path>***/HDFS/component_env** to change the memory upper limit of the Yarn client. For example, if you want to set the maximum memory to 1 GB, run the following command:

   ```
   export CLIENT_GC_OPTS="-Xmx1G"
   ```

   For clusters earlier than MRS 3.x: You can modify **GC_OPTS_YARN** in *< Client installation path >***/HDFS/component_env** to change the memory upper limit of the Yarn client. For example, if you want to set the maximum memory to 1 GB, run the following command:

   ```
   export GC_OPTS_YARN="-Xmx1G"
   ```

   After the modification, run the following command to make the modification take effect:

   source <*Client installation path*>/**/bigdata_env**

2. How Can I Set the Log Level When the Yarn Client Is Running?

   By default, the logs generated during the running of the Yarn client are printed to the console. The default log level is INFO. To enable the DEBUG log level for fault locating, run the following command to export an environment variable:

export YARN_ROOT_LOGGER=DEBUG,console

Then run the Yarn Shell command to print DEBUG logs.

If you want to print INFO logs again, run the following command:

export YARN_ROOT_LOGGER=INFO,console

12.27.4 Configuring Resources for a NodeManager Role Instance

Scenario

If the hardware resources (such as the number of CPU cores and memory size) of the nodes for deploying NodeManagers are different but the NodeManager available hardware resources are set to the same value, the resources may be wasted or the status may be abnormal. You need to change the hardware resource configuration for each NodeManager to ensure that the hardware resources can be fully utilized.

Impact on the System

NodeManager role instances must be restarted for the new configuration to take effect, and the role instances are unavailable during restart.

Prerequisites

- For versions earlier than MRS 3.x: You have logged in to the MRS management console.
- Clusters of MRS 3.x or later: You have logged in to Manager.

Procedure

For versions earlier than MRS 3.*x*, perform the following operations:

Step 1 Choose **Clusters** > **Active Clusters**, and click a cluster name. Choose **Components** > **Yarn** > **Instances**.

Step 2 Click **NodeManager** in the **Role** column and go to the **Instance Configuration** tab page. Select **All** from the **Basic** drop-down list, and search for the required parameters.

Step 3 Enter **yarn.nodemanager.resource.cpu-vcores** in the search box, and set the number of vCPUs that can be used by NodeManager on the current node. You are advised to set this parameter to 1.5 to 2 times the number of actual logical CPUs on the node. Enter **yarn.nodemanager.resource.memory-mb** in the search box, and set the physical memory size that can be used by NodeManager on the current node. You are advised to set this parameter to 75% to 90% of the actual physical memory size of the node.

NOTE

Enter **yarn.scheduler.maximum-allocation-vcores** in the search box, and set the maximum number of available CPUs in a container. Enter **yarn.scheduler.maximum-allocation-mb** in the search box, and set the maximum available memory of a container. The instance level cannot be changed. The parameter values need to be changed in the configuration of the Yarn service, and the Yarn service needs to be restarted for the changes to take effect.

Step 4 Click **Save Configuration**, select **Restart the affected services or instances**, and click **OK** to restart the NodeManager role instance.

Operation succeeded is displayed. Click **Finish**. The NodeManager role instance is started successfully.

----End

For MRS 3.*x* or later, perform the following operations:

Step 1 Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Instance**.

Step 2 Click the role instance name corresponding to the node where NodeManager is deployed, switch to **Instance Configuration**, and select **All Configurations**.

Step 3 Enter **yarn.nodemanager.resource.cpu-vcores** in the search box, and set the number of vCPUs that can be used by NodeManager on the current node. You are advised to set this parameter to 1.5 to 2 times the number of actual logical CPUs on the node. Enter **yarn.nodemanager.resource.memory-mb** in the search box, and set the physical memory size that can be used by NodeManager on the current node. You are advised to set this parameter to 75% of the actual physical memory size of the node.

NOTE

Enter **yarn.scheduler.maximum-allocation-vcores** in the search box, and set the maximum number of available CPUs in a container. Enter **yarn.scheduler.maximum-allocation-mb** in the search box, and set the maximum available memory of a container. The instance level cannot be changed. The parameter values need to be changed in the configuration of the Yarn service, and the Yarn service needs to be restarted for the changes to take effect.

Step 4 Click **Save**, and then click **OK**. to restart the NodeManager role instance.

A message is displayed, indicating that the operation is successful. Click **Finish**. The NodeManager role instance is started successfully.

----End

12.27.5 Changing NodeManager Storage Directories

Scenario

If the storage directories defined by the Yarn NodeManager are incorrect or the Yarn storage plan changes, the system administrator needs to modify the NodeManager storage directories on Manager to ensure that the Yarn works properly. The storage directories of NodeManager include the local storage directory **yarn.nodemanager.local-dirs** and log directory **yarn.nodemanager.log-dirs**. Changing the ZooKeeper storage directory includes the following scenarios:

- Change the storage directory of the NodeManager role. In this way, the storage directories of all NodeManager instances are changed.
- Change the storage directory of a single NodeManager instance. In this way, only the storage directory of this instance is changed, and the storage directories of other instances remain the same.

Impact on the System

- The cluster needs to stopped and restarted during the process of changing the storage directory of the NodeManager role, and the cluster cannot provide services before started.
- The NodeManager instance needs to stopped and restarted during the process of changing the storage directory of the instance, and the instance at this node cannot provide services before it is started.
- The directory for storing service parameter configurations must also be updated.
- After the storage directories of NodeManager are changed, you need to download and install the client again.

Prerequisites

- New disks have been prepared and installed on each data node, and the disks are formatted.
- New directories have been planned for storing data in the original directories.
- The system administrator account **admin** has been prepared.

Procedure

For versions earlier than MRS 3.*x*, perform the following operations:

Step 1 Check the environment.

1. Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click a cluster name. Choose **Components** and check whether health status of Yarn is **Good**.
   - If yes, go to **Step 1.3**.
   - If no, the Yarn status is unhealthy. Go to **Step 1.2**.
2. Rectify the Yarn fault. No further action is required.
3. Determine whether to change the storage directory of the NodeManager role or that of a single NodeManager instance:
   - To change the storage directory of the NodeManager role, go to **Step 2**.
   - To change the storage directory of a single NodeManager instance, go to **Step 3**.

Step 2 Change the storage directory of the NodeManager role.

1. Choose **Clusters** > **Active Clusters**, and click a cluster name. Choose **Components** > **Yarn** > **Stop** to stop the Yarn service.
2. Log in to the ECS server and go to each node where Yarn is installed as user **root**. Perform the following operations:
   a. Create a target directory.

      For example, to create the target directory **${BIGDATA_DATA_HOME}/data2**, run the following command:

      mkdir ${BIGDATA_DATA_HOME}/data2

   b. Mount the target directory to the new disk.

      For example, mount **${BIGDATA_DATA_HOME}/data2** to the new disk.

c. Modify permissions on the new directory.

For example, to modify permissions on the **${BIGDATA_DATA_HOME}/data2** directory, run the following commands:

chmod 750 ${BIGDATA_DATA_HOME}/data2 -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/data2 -R**

3. On the MRS console, choose **Clusters** > **Active Clusters** and click a cluster name. Choose **Components** > **Yarn** > **Instances**. Select the NodeManager instance of the corresponding host. Choose **Instance Configuration** > **All Configurations**.

Change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to the new target directory.

For example, change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to **/srv/BigData/data2/nm/containerlogs**.

4. Click **Save Configuration**, select **Restart the affected services or instances**, and click **OK** Restart the Yarn service.

Click **Finish** when the system displays "Operation successful". Yarn is successfully started. No further action is required.

Step 3 Change the storage directory of a single NodeManager instance.

1. Choose **Clusters** > **Active Clusters**, and click a cluster name. Choose **Components** > **Yarn** > **Instances**. Select the NodeManager instance whose storage directory needs to be modified, and choose **More** > **Stop Instance**.
2. Log in to the ECS and go to the NodeManager node as user **root**. Perform the following operations:

a. Create a target directory.

For example, to create the target directory **${BIGDATA_DATA_HOME}/data2**, run the following command:

mkdir ${BIGDATA_DATA_HOME}/data2

b. Mount the target directory to the new disk.

For example, mount **${BIGDATA_DATA_HOME}/data2** to the new disk.

c. Modify permissions on the new directory.

For example, to modify permissions on the **${BIGDATA_DATA_HOME}/data2** directory, run the following commands:

chmod 750 ${BIGDATA_DATA_HOME}/data2 -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/data2 -R**

3. On the MRS console, click the specified NodeManager instance and switch to the **Instance Configuration** tab page.

Change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to the new target directory.

For example, change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to **/srv/BigData/data2/nm/containerlogs**.

4. Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the NodeManager instance.

Click **Finish** when the system displays "Operation successful". The NodeManager instance is successfully started.

----End

For MRS 3.*x* or later, perform the following operations:

Step 1 Check the environment.

1. Log in to Manager, choose **Cluster** > *Name of the desired cluster* > **Service** to check whether **Running Status** of Yarn is **Normal**.
   - If yes, go to **1.c**.
   - If no, the Yarn status is unhealthy. In this case, go to **1.b**.
2. Rectify faults of Yarn. No further action is required.
3. Determine whether to change the storage directory of the NodeManager role or that of a single NodeManager instance:
   - To change the storage directory of the NodeManager role, go to **2**.
   - To change the storage directory of a single NodeManager instance, go to **3**.

Step 2 Change the storage directory of the NodeManager role.

1. Choose **Cluster** > *Name of the desired cluster* > **Service** > **Yarn** > **Stop** to stop the Yarn service.
2. Log in to each data node where the Yarn service is installed as user **root** and perform the following operations:
   a. Create a target directory.

      For example, to create the target directory **${BIGDATA_DATA_HOME}/data2**, run the following command:

      mkdir ${BIGDATA_DATA_HOME}/data2

   b. Mount the target directory to the new disk.

      For example, mount **${BIGDATA_DATA_HOME}/data2** to the new disk.

   c. Modify permissions on the new directory.

      For example, to modify permissions on the **${BIGDATA_DATA_HOME}/data2** directory, run the following commands:

      chmod 750 ${BIGDATA_DATA_HOME}/data2 -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/data2 -R**
3. On the Manager portal, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Instance**. Select the NodeManager instance of the corresponding host, click **Instance Configuration**, and select **All Configurations**.

   Change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to the new target directory.

   For example, change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to **/srv/BigData/data2/nm/containerlogs**.
4. Click **Save**, and then click **OK**. Restart the Yarn service.

   Click **Finish** when the system displays "Operation successful". Yarn is successfully started. No further action is required.

Step 3 Change the storage directory of a single NodeManager instance.

1. Choose **Cluster** > *Name of the desired cluster* > **Service** > **Yarn** > **Instance**, select the NodeManager instance whose storage directory needs to be modified, and choose **More** > **Stop**.

2. Log in to the NodeManager node as user **root**, and perform the following operations:
   a. Create a target directory.
      For example, to create the target directory **${BIGDATA_DATA_HOME}/data2**, run the following command:
      mkdir ${BIGDATA_DATA_HOME}/data2
   b. Mount the target directory to the new disk.
      For example, mount **${BIGDATA_DATA_HOME}/data2** to the new disk.
   c. Modify permissions on the new directory.
      For example, to modify permissions on the **${BIGDATA_DATA_HOME}/data2** directory, run the following commands:
      chmod 750 ${BIGDATA_DATA_HOME}/data2 -R and **chown omm:wheel ${BIGDATA_DATA_HOME}/data2 -R**
3. On Manager, click the specified NodeManager instance, and switch to the **Instance Configuration** page.
   Change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to the new target directory.
   For example, change the value of **yarn.nodemanager.local-dirs** or **yarn.nodemanager.log-dirs** to **/srv/BigData/data2/nm/containerlogs**.
4. Click **Save**, and then click **OK** to restart the NodeManager instance.
   Click **Finish** when the system displays "Operation successful". The NodeManager instance is successfully started.

----End

12.27.6 Configuring Strict Permission Control for Yarn

Scenario

In the multi-tenant scenario in security mode, a cluster can be used by multiple users, and tasks of multiple users can be submitted and executed. Users are invisible to each other. A permission control mechanism is required to prevent task information of users from being obtained by other users.

For example, if user B logs in to the system and views the application list when the application submitted by user A is running, user B should not be able to view the application information of user A.

Configuration Description

- Viewing Yarn configuration parameters
  Go to the **All Configurations** page of Yarn and enter a parameter name list in **Table 12-451** in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-451 Parameter description

Parameter	Description	Default Value
yarn.acl.enable	Whether to enable Yarn permission control	true
yarn.webapp.filter-entity-list-by-user	Whether to enable the strict view function. After this function is enabled, a login user can view only the content that the user has the permission to view. To enable this function, set **yarn.acl.enable** to **true**. **NOTE** This parameter applies to clusters of MRS 3.*x* or later.	true

- Viewing MapReduce configuration parameters

  Go to the **All Configurations** page of MapReduce and enter a parameter name in **Table 12-452** in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

  Table 12-452 Parameter description

Parameter	Description	Default Value
mapreduce.cluster.acls.enabled	Whether to enable permission control of MapReduce JobHistoryServer This parameter is a client parameter and takes effect after permission control is enabled on the JobHistoryServer server.	true
yarn.webapp.filter-entity-list-by-user	Whether to enable the strict view of MapReduce JobHistoryServer. After the strict view is enabled, a login user can view only the content that the user has the permission to view. This parameter is a server parameter of JobHistoryServer. It indicates that permission control is enabled for JHS. However, whether to control a specific application is determined by the client parameter **mapreduce.cluster.acls.enabled**. **NOTE** This parameter applies to clusters of MRS 3.*x* or later.	true

NOTICE

The preceding configurations affect the RESTful API and Shell command results. After the preceding configurations are enabled, the return results of RESTful API calls and shell commands contain only the information that the user has the permission to view.

If **yarn.acl.enable** or **mapreduce.cluster.acls.enabled** is set to **false**, the Yarn or MapReduce permission verification function is disabled. In this case, any user can submit tasks and view task information on Yarn or MapReduce, which poses security risks. Exercise caution when performing this operation.

12.27.7 Configuring Container Log Aggregation

Scenario

Yarn provides the container log aggregation function to collect logs generated by containers on each node to HDFS to release local disk space. You can collect logs in either of the following ways:

- After the application is complete, collect container logs to HDFS at a time.
- During application running, periodically collect log segments generated by containers and save them to HDFS.

Configuration Description

Navigation path for setting parameters:

Go to the **All Configurations** page of Yarn and enter a parameter name list in **Table 12-453** in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

The **yarn.nodemanager.remote-app-log-dir-suffix** parameter must be configured on the Yarn client. The configurations on the ResourceManager, NodeManager, and JobHistory nodes must be the same as those on the Yarn client.

The periodic log collection function applies only to MapReduce applications, for which rolling output of log files must be configured. **Table 12-455** describes the configurations in the **mapred-site.xml** configuration file on the MapReduce client node.

Table 12-453 Parameter description

Parameter	Description	Default Value
yarn.log-aggregation-enable	Whether to enable container log aggregation ● If this parameter is set to **true**, logs are collected to the HDFS directory. ● If this parameter is set to **false**, the function is disabled, and logs are not collected to HDFS. After changing the parameter value, restart the Yarn service for the setting to take effect. **NOTE** ● The container logs that are generated before the parameter is set to **false** and the setting takes effect cannot be obtained from the web UI. ● If you want to view the logs generated before on the web UI, you are advised to set this parameter to **true**.	true
yarn.nodemanager.log-aggregation.roll-monitoring-interval-seconds	Interval for NodeManager to periodically collect logs ● If this parameter is set to **-1** or **0**, periodic log collection is disabled. Logs are collected at a time after application running is complete. ● The minimum collection interval can be set to 3,600 seconds. If this parameter is set to a value greater than 0 and less than 3,600, the collection interval is 3,600 seconds. Interval for NodeManager to wake up and upload logs. If this parameter is set to **-1** or **0**, rolling monitoring is disabled and logs are aggregated when the application task is complete. The value must be greater than or equal to **-1**.	-1

Parameter	Description	Default Value
yarn.nodemanager.disk-health-checker.log-dirs.max-disk-utilization-per-disk-percentage	Maximum percentage of the Yarn disk quota that can be occupied by the container log directory on each disk. When the space occupied by the log directory exceeds the value of this parameter, the periodic log collection service is triggered to start a log collection activity beyond the period to release the local disk space. Maximum space for container logs that can be provided on each disk. If the disk space occupied by container logs exceeds this threshold, data aggregation in rolling mode is triggered. ● For clusters of versions earlier than MRS 3.x: The valid value range of the maximum disk quota percentage is 0 to 100. If the value is less than or equal to **0**, it is forcibly reset to **25**. If the value is greater than 100, the value is forcibly reset to **25**. ● For clusters of MRS 3.*x* or later: The valid value range of the maximum disk quota percentage is –1 to 100. If the value is less than –1, it is forcibly reset to **25**. If the value is greater than 100, the value is forcibly reset to **25**. If you set the value to **–1**, the disk capacity detection function for Container log directory is disabled. **NOTE** ● Percentage of the available disk space of the container log directory = Percentage of the available disk space of Yarn (**yarn.nodemanager.disk-health-checker.max-disk-utilization-per-disk-percentage**) x Percentage of the available disk space of the container log directory (**yarn.nodemanager.disk-health-checker.log-dirs.max-disk-utilization-per-disk-percentage**) ● Only applications with the periodic log collection function enabled can trigger log collection when the disk quota of the log directory exceeds the threshold.	25

Parameter	Description	Default Value
yarn.nodemanager.remote-app-log-dir-suffix	Name of the HDFS folder in which container logs are to be stored. This parameter and **yarn.nodemanager.remote-app-log-dir** form the full path for storing container logs. That is, **{yarn.nodemanager.remote-app-log-dir}/${user}/{yarn.nodemanager.remote-app-log-dir-suffix}**. **NOTE** *{user}* indicates the username for running the task.	logs
yarn.nodemanager.log-aggregator.on-fail.remain-log-in-sec	Duration for retaining container logs on the local host after the logs fail to be collected, in second • If this parameter is set to **0**, local logs are deleted immediately. • If this parameter is set to a positive number, local logs are retained for this period.	604800

Go to the **All Configurations** page of MapReduce and enter a parameter name in **Table 12-454** in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-454 Parameter description

Parameter	Description	Default Value
yarn.log-aggregation.retain-seconds	Duration for retaining aggregated logs, in second • If this parameter is set to **–1**, the container logs will be retained permanently in the HDFS. • If this parameter is set to **0** or a positive integer, container logs will be stored for such a period and deleted after the period expires. **NOTE** A short period may increase load of the NameNode. Therefore, you are advised to set this parameter to a proper value.	1296000

Parameter	Description	Default Value
yarn.log-aggregation.retain-check-interval-seconds	Interval for storing container logs in HDFS, in second • If this parameter is set to **-1** or **0**, the interval will be one tenth of the period specified by **yarn.log-aggregation.retain-seconds**. **NOTE** If this parameter is set to **-1** or **0**, **yarn.log-aggregation.retain-seconds** cannot be set to **0**. • If this parameter is set to a positive number, container logs in HDFS will be scanned at such an interval. **NOTE** A short interval may increase load of the NameNode. Therefore, you are advised to set this parameter to a proper value.	86400

Go to the **All Configurations** page of Yarn and enter a parameter name list in **Table 12-455** in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-455 Configuring rolling output of MapReduce application log files

Parameter	Description	Default Value
mapreduce.task.userlog.limit.kb	Maximum size of a single task log file of the MapReduce application. When the maximum size of the log file has been reached, a new log file is generated. The value **0** indicates that the size of the log file is not limited.	51200

Parameter	Description	Default Value
yarn.app.mapreduce.task.container.log.backups	Maximum number of task logs that can be retained for the MapReduce application. If this parameter is set to **0**, rolling output is disabled. Number of task log backup files when ContainerRollingLogAppender (CRLA) is used. By default, ContainerLogAppender (CLA) is used and container logs are not rolled back. When both **mapreduce.task.userlog.limit.kb** and **yarn.app.mapreduce.task.container.log.backups** are greater than 0, CRLA is enabled. The value ranges from 0 to 999.	10
yarn.app.mapreduce.am.container.log.limit.kb	Maximum size of a single ApplicationMaster log file of the MapReduce application, in KB. When the maximum size of the log file has been reached, a new log file is generated. The value **0** indicates that the size of a single ApplicationMaster log file is not limited.	51200
yarn.app.mapreduce.am.container.log.backups	Maximum number of ApplicationMaster logs that can be retained for the MapReduce application. If this parameter is set to **0**, rolling output is disabled. Number of ApplicationMaster log backup files when CRLA is used. By default, CLA is used and container logs are not rolled back. When both **yarn.app.mapreduce.am.container.log.limit.kb** and **yarn.app.mapreduce.am.container.log.backups** are greater than 0, CRLA is enabled for the ApplicationMaster. The value ranges from 0 to 999.	20
yarn.app.mapreduce.shuffle.log.backups	Maximum number of shuffle logs that can be retained for the MapReduce application. If this parameter is set to **0**, rolling output is disabled. When both **yarn.app.mapreduce.shuffle.log.limit.kb** and **yarn.app.mapreduce.shuffle.log.backups** are greater than 0, **syslog.shuffle** uses CRLA. The value ranges from 0 to 999.	10

Parameter	Description	Default Value
yarn.app.mapreduce.shuffle.log.limit.kb	Maximum size of a single shuffle log file of the MapReduce application, in KB. When the maximum size of the log file has been reached, a new log file is generated. If this parameter is set to **0**, the size of a single shuffle log file is not limited. The value must be greater than or equal to **0**.	51200

12.27.8 Using CGroups with YARN

This section applies to clusters of MRS 3.*x* or later.

Scenario

CGroups is a Linux kernel feature. In YARN this feature allows containers to be limited in their resource usage (example, CPU usage). Without CGroups, it is hard to limit the container CPU usage. Without CGroups, it is hard to limit the container CPU usage.

NOTE

Currently, CGroups is only used for limiting the CPU usage.

Configuration Description

For details about how to configure the CGroups function for CPU isolation and security, see the Hadoop official website: **http://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-site/NodeManagerCgroups.html**

CGroups is a Linux kernel feature and is enabled by using LinuxContainerExecutor. For details about how to configure the LinuxContainerExecutor for security, see the official website. You can learn the file system permissions assigned for users and user groups from documentation published on the official website. For details, see **http://hadoop.apache.org/docs/r3.1.1/hadoop-project-dist/hadoop-common/SecureMode.html#LinuxContainerExecutor**.

NOTE

- Do not modify users, user groups, and related permissions of various paths in the corresponding file system. Otherwise, functions of CGroups may become abnormal.
- If the parameter value of **yarn.nodemanager.resource.percentage-physical-cpu-limit** is too small, the number of available cores may be less than one. For example, if the parameter of a four-core node is set to 20%, the number available core is less than one. As a result, all cores will be used. The Quota mode can be used in Linux versions, for example, Cent OS, that do not support Quota mode.

The table below describes the parameter for configuring cpuset mode, that is, only configured CPUs can be used by YARN.

Table 12-456 Parameter description

Parameter	Description	Default Value
yarn.nodemanager.linux-container-executor.cgroups.cpu-set-usage	Whether to enable the cpuset mode. If this parameter is set to **true**, the cpuset mode is enabled.	false

The table below describes the parameters for configuring the strictcpuset mode, that is, only configured CPUs can be used by containers.

Table 12-457 Parameter description

Parameter	Description	Default Value
yarn.nodemanager.linux-container-executor.cgroups.cpu-set-usage	Whether to enable the cpuset mode. If this parameter is set to **true**, the cpuset mode is enabled.	false
yarn.nodemanager.linux-container-executor.cgroups.cpuset.strict.enabled	Whether containers use allocated CPUs. If this parameter is set to **true**, the container can use the allocated CPUs.	false

To switch from cpuset mode to quota mode, the following conditions must be met:

- Set the **yarn.nodemanager.linux-container-executor.cgroups.cpu-set-usage** parameter to **false**.
- Delete container folders if exists.
- Delete all the CUPs configured in the **cpuset.cpus** file.

Procedure

Step 1 Log in to Manager. Choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Configurations** and select **All Configurations**.

Step 2 In the navigation pane on the left, choose **NodeManager** > **Customization** and find the **yarn-site.xml** file.

Step 3 Add the parameters in **Table 12-456** and **Table 12-457** as user-defined parameters.

Based on the configuration files and parameter functions, locate the row where parameter **yarn-site.xml** resides. Enter the parameter name in the **Name** column and enter the parameter value in the **Value** column.

Click **+** to add a customized parameter.

Step 4 Click **Save**. In the displayed **Save Configuration** dialog box, confirm the modification and click **OK**. Click **Finish** when the system displays "Operation succeeded". The configuration is successfully saved.

After the configuration is saved, restart the Yarn service whose configuration has expired for the configuration to take effect.

----End

12.27.9 Configuring the Number of ApplicationMaster Retries

Scenario

When resources are insufficient or ApplicationMaster fails to start, a client probably encounters running errors.

Configuration Description

Go to the **All Configurations** page of Yarn and enter a parameter name list in **Table 12-458** in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-458 Parameter description

Parameter	Description	Default Value
yarn.resourcemanager.am.max-attempts	Number of retries of the ApplicationMaster. Increasing the number of retries can prevent ApplicationMaster startup failures caused by insufficient resources. This applies to global settings of all ApplicationMasters. Each ApplicationMaster can use an API to set an independent maximum number of retries. However, the number of retries cannot be greater than the global maximum number of retries. If the value is greater than the global maximum number of retries, the ResourceManager overwrites the value to allow at least one retry. The value must be greater than or equal to **1**.	5

12.27.10 Configure the ApplicationMaster to Automatically Adjust the Allocated Memory

This section applies to clusters of MRS 3.*x* or later.

Scenario

During the process of starting the configuration, when the ApplicationMaster creates a container, the allocated memory is automatically adjusted according to

the total number of tasks, which makes resource utilization more flexible and improves the fault tolerance of the client application.

Configuration Description

Navigation path for setting parameters:

On Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Yarn** > **Configuration**. On the displayed page, select **All Configurations** and enter **mapreduce.job.am.memory.policy**.

Configuration description

If the default value of the parameter is left empty. In this case, the automatic adjustment policy is not enabled. The memory of ApplicationMaster is still affected by the value of **yarn.app.mapreduce.am.resource.mb**.

The value of **mapreduce.job.am.memory.policy** consists of five items, and they are separated by colons (:) and commas (,) in the following format: **baseTaskCount:taskStep:memoryStep,minMemory:maxMemory**. The format is strictly checked when the value is entered.

Table 12-459 Parameter description

Parameter	Description	Setting Requirement
baseTaskCount	Indicates the total number of tasks. The configuration of ApplicationMaster is valid only when the total number of tasks (on the sum of the Map and Reduce ends) is greater than or equal to the value of this parameter.	The value cannot be empty and must be greater than 0.
taskStep	Indicates the incremental step length of tasks. This parameter and **memoryStep** determine the memory adjustment amount.	The value cannot be empty and must be greater than 0.
memoryStep	Indicates the incremental memory step. The memory capacity is increased based on the value of **yarn.app.mapreduce.am.resource.mb**.	The value cannot be empty and must be greater than 0. The unit is MB.
minMemory	Indicates the lower limit of the memory that can be automatically adjusted. If the memory after the automatic adjustment is less than or equal to the value of this parameter, the value of **yarn.app.mapreduce.am.resource.mb** is used.	The value cannot be empty. It must be greater than 0 and cannot be greater than the value of **maxMemory**. Unit: MB

Parameter	Description	Setting Requirement
maxMemory	Indicates the upper limit of memory that can ve automatically adjusted. If the adjusted memory exceeds the upper limit, use this value as the final value.	The value cannot be empty. It must be greater than 0 and cannot be less than the value of **minMemory**. Unit: MB

Example Value

Configuration:

- yarn.app.mapreduce.am.resource.mb=1536
- mapreduce.job.am.memory.policy=100:10:50,1200:2000
- Total number of tasks of an application =120

The calculation process is as follows:

Memory after adjustment = 1536 + [(120 – 100)/10] x 50 = 1636. In this example, memory after adjustment 1636 is greater than the value of **minMemory 1200**, and less than the value of **maxMemory 2000**. Therefore, the ApplicationMaster memory is set to **1636 MB**.

If the value of **memStep** is changed to **250**, the calculation formula is as follows: Memory after adjustment = 1536 + [(120 – 100) / 10] x 250 = 2136. In this case, the memory after adjustment is greater than the value of **maxMemory 2000**. As a result, the value of **ApplicationMaster** is set to **2000 MB**.

NOTE

If the memory after adjustment is lower than the value of **minMemory**, the configuration does not take effect but the value is still printed on the backend server. This value is provided as the reference for adjusting the value of **minMemory**.

12.27.11 Configuring the Access Channel Protocol

Scenario

The value of the **yarn.http.policy** parameter must be consistent on both the server and clients. Web UIs on clients will be garbled if an inconsistency exists, for example, the parameter value is **HTTPS_ONLY** on the server but it is left unspecified on a client (the parameter value **HTTP_ONLY** is applied to the client by default). Set the **yarn.http.policy** parameters on the clients and server to prevent garbled characters from being displayed on the clients.

Procedure

Step 1 On Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Configurations**. On the displayed page, select **All Configurations** and enter **yarn.http.policy**.

- In security mode, set this parameter to **HTTPS_ONLY**.
- In normal mode, set this parameter to **HTTP_ONLY**.

Step 2 Log in to the node where the client is installed as the client installation user.

Step 3 Run the following command to switch to the client installation directory:

cd /opt/client

Step 4 Run the following command to edit the **yarn-site.xml** file:

vi Yarn/config/yarn-site.xml

Change the value of **yarn.http.policy**.

In security mode, set this parameter to **HTTPS_ONLY**.

In normal mode, set this parameter to **HTTP_ONLY**.

Step 5 Run the **:wq** command to save execution.

Step 6 Restart the client for the settings to take effect.

----End

12.27.12 Configuring Memory Usage Detection

Scenario

If memory usage of the submitted application cannot be estimated, you can modify the configuration on the server to determine whether to check the memory usage.

If the memory usage is not checked, the container occupies the memory until the memory overflows. If the memory usage exceeds the configured memory size, the corresponding container is killed.

Configuration Description

Go to the **All Configurations** page of Yarn and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-460 Parameter description

Parameter	Description	Default Value
yarn.nodemanager.vmem-check-enabled	Whether to enable virtual memory usage detection. If the memory used by a task exceeds the allocated memory size, the task is forcibly stopped. • If the value is **true**, the virtual memory will be checked. • If the value is **false**, the virtual memory will not be checked.	For versions earlier than MRS 3.x: false For MRS 3.x or later: true
yarn.nodemanager.pmem-check-enabled	Whether to enable physical memory usage detection. If the memory used by a task exceeds the allocated memory size, the task is forcibly stopped. • If the value is **true**, the physical memory will be checked. • If the value is **false**, the physical memory will not be checked.	true

12.27.13 Configuring the Additional Scheduler WebUI

Scenario

If the custom scheduler is set in ResourceManager, you can set the corresponding web page and other Web applications for the custom scheduler.

Configuration Description

Go to the **All Configurations** page of Yarn and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-461 Configuring the Additional Scheduler WebUI

Parameter	Description	Default Value
hadoop.http.rmwebapp.scheduler.page.classes	Load the corresponding web page for the custom scheduler on the RM WebUI. This parameter is valid only when **yarn.resourcemanager.scheduler.class** is set to a custom scheduler.	-
yarn.http.rmwebapp.external.classes	Load the custom web application in the RM Web service.	-

12.27.14 Configuring Yarn Restart

Scenario

The Yarn Restart feature includes ResourceManager Restart and NodeManager Restart.

- When ResourceManager Restart is enabled, the new active ResourceManager node loads the information of the previous active ResourceManager node, and takes over container status information on all NodeManager nodes to continue service running. In this way, status information can be saved by periodically executing checkpoint operations, avoiding data loss.
- When NodeManager Restart is enabled, NodeManager locally saves information about containers running on the node. After NodeManager is restarted, the container running progress on the node will not be lost by restoring the saved status information.

Configuration Description

Go to the **All Configurations** page of Yarn and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Configure ResourceManager Restart as follows:

Table 12-462 Parameter description of ResourceManager Restart

Parameter	Description	Default Value
yarn.resourcemanager.recovery.enabled	Whether to enable ResourceManager to restore the status after startup. If this parameter is set to **true**, **yarn.resourcemanager.store.class** must also be set.	true
yarn.resourcemanager.store.class	State-store class used to store the application and task statuses and certificate content.	For clusters of versions earlier than MRS 3.x: **org.apache.hadoop.yarn.server.resourcemanager.recovery.ZKRMStateStore** For clusters of MRS 3.*x* or later: org.apache.hadoop.yarn.server.resourcemanager.recovery.AsyncZKRMStateStore
yarn.resourcemanager.zk-state-store.parent-path	Directory for storing ZKRMStateStore in ZooKeeper	/rmstore

Parameter	Description	Default Value
yarn.resourcemanager.work-preserving-recovery.enabled	Whether to enable ResourceManager work serving. This configuration is used only for Yarn feature verification.	true
yarn.resourcemanager.state-store.async.load	Whether to apply asynchronous restoration to completed applications.	For clusters of versions earlier than MRS 3.x: **false** For MRS 3.x or later: **true**
yarn.resourcemanager.zk-state-store.num-fetch-threads	If asynchronous restoration is enabled, increasing the number of working threads can speed up the restoration of task information stored in ZooKeeper. The value must be greater than 0.	For clusters of versions earlier than MRS 3.x: **1** For MRS 3.x or later: **20**

Configure NodeManager Restart as follows:

Table 12-463 Parameter description of NodeManager Restart

Parameter	Description	Default Value
yarn.nodemanager.recovery.enabled	Whether to enable the function of collecting logs upon a log collection failure when NodeManager is restarted and whether to restore the unfinished application	true
yarn.nodemanager.recovery.dir	Local directory used by NodeManager to store container status It applies to clusters of MRS 3.*x* or later.	${SRV_HOME}/tmp/yarn-nm-recovery
yarn.nodemanager.recovery.supervised	Whether NodeManager is monitored. After this parameter is enabled, NodeManager does not clear containers after exit. NodeManager assumes that it will restart and restore containers immediately.	true

12.27.15 Configuring ApplicationMaster Work Preserving

This section applies to clusters of MRS 3.*x* or later.

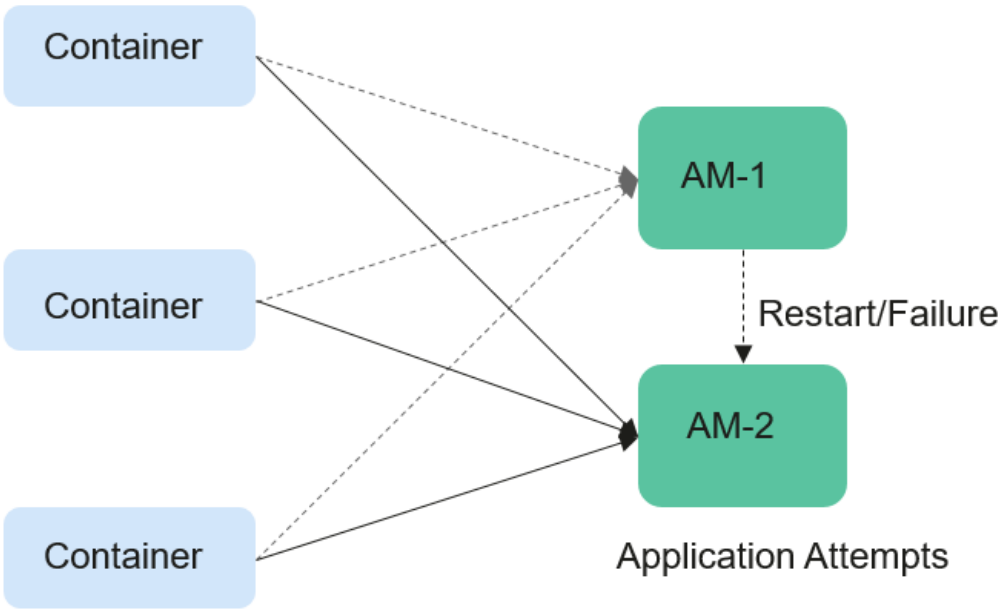
Scenario

In YARN, ApplicationMasters run on NodeManagers just like every other container (ignoring unmanaged ApplicationMasters in this context). ApplicationMasters may break down, exit, or shut down. If an ApplicationMaster node goes down, ResourceManager kills all the containers of ApplicationAttempt, including containers running on NodeManager. ResourceManager starts a new ApplicationAttempt node on another compute node.

For different types of applications, we want to handle ApplicationMaster restart events in different ways. MapReduce applications aim to prevent task loss but allow the loss of the currently running container. However, for the long-period YARN service, users may not want the service to stop due to the ApplicationMaster fault.

YARN can retain the status of the container when a new ApplicationAttempt is started. Therefore, running jobs can continue to operate without faults.

Figure 12-74 ApplicationMaster job preserving



Configuration Description

Go to the **All Configurations** page of Yarn and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Set the following parameters based on **Table 12-464**.

Table 12-464 Parameter description

Parameter	Description	Default Value
yarn.app.mapreduce.am.work-preserve	Whether to enable the ApplicationMaster job retention feature.	false
yarn.app.mapreduce.am.umbilical.max.retries	Maximum number of attempts to restore a running container in the ApplicationMaster job retention feature.	5
yarn.app.mapreduce.am.umbilical.retry.interval	Specifies the interval at which a running container attempts to recover in the ApplicationMaster job retention feature. Unit: millisecond	10000
yarn.resourcemanager.am.max-attempts	The number of retries of ApplicationMaster. Increasing the number of retries prevents ApplicationMaster startup failures caused by insufficient resources. This applies to global settings of all ApplicationMasters. Each ApplicationMaster can use an API to set an independent maximum number of retries. However, the number of retries cannot be greater than the global maximum number of retries. If the value is greater than the global maximum number of retries, the ResourceManager overwrites the value The value must be greater than or equal to **1**.	2

12.27.16 Configuring the Localized Log Levels

This section applies to clusters of MRS 3.*x* or later.

Scenarios

The default log level of localized container is **INFO**. You can change the log level by configuring **yarn.nodemanager.container-localizer.java.opts**.

Configuration Description

On Manager, choose **Cluster** > *Name of the desired cluster* > **Service** > **Yarn** > **Configuration**. Select **All Configurations** and set the following parameters in the configuration file **yarn-site.xml** of NodeManager to change the log level.

Table 12-465 Parameter description

Parameter	Description	Default Value
yarn.nodemanager.container-localizer.java.opts	The additional **jvm** parameters are provided for the localized container process.	-Xmx256m -Djava.security.krb5.conf=${KRB5_CONFIG}

The default value is **-Xmx256m -Djava.security.krb5.conf=${KRB5_CONFIG}** and the default log level is info. To change the localized log level of the container, add the following content:

```
-Dhadoop.root.logger=<LOG_LEVEL>,localizationCLA
```

Example:

To change the local log level to **DEBUG**, set the parameter as follows:

```
-Xmx256m -Dhadoop.root.logger=DEBUG,localizationCLA
```

NOTE

Allowed log levels are as follows: FATAL, ERROR, WARN, INFO, DEBUG, TRACE, and ALL.

12.27.17 Configuring Users That Run Tasks

This section applies to clusters of MRS 3.*x* or later.

Scenario

Currently, YARN allows the user that starts the NodeManager to run the task submitted by all other users, or the users to run the task submitted by themselves.

Configuration Description

On Manager, choose **Cluster** > *Name of the desired cluster* > **Services** > **Yarn** > **Configurations**. Click **All Configurations** Enter a parameter name in the search box.

Table 12-466 Parameter description

Parameter	Description	Default Value
yarn.nodemanager.linux-container-executor.user	Indicates the user who runs a task.	The value is left blank by default. **NOTE** The value is left blank by default. The user who submits a task is the actual person who runs the task.
yarn.nodemanager.container-executor.class	Indicates the executor who starts a task.	org.apache.hadoop.yarn.server.nodemanager.EnhancedLinuxContainerExecutor

NOTE

- Set **yarn.nodemanager.linux-container-executor.user** to configure the user who runs the container. This parameter is left blank by default. The user who submits the task is the person who runs the container. This parameter is valid only when **yarn.nodemanager.container-executor.class** is set to **org.apache.hadoop.yarn.server.nodemanager.EnhancedLinuxContainerExecutor**.
- In non-security mode, if **yarn.nodemanager.linux-container-executor.user** is set to **omm**, **yarn.nodemanager.linux-container-executor.nonsecure-mode.local-user** must also be set to **omm**.
- For security reasons, it is advised to retain the default values of **yarn.nodemanager.linux-container-executor.user** and **yarn.nodemanager.container-executor.class**.

12.27.18 Yarn Log Overview

Log Description

The default paths for saving Yarn logs are as follows:

- ResourceManager: **/var/log/Bigdata/yarn/rm** (run logs) and **/var/log/Bigdata/audit/yarn/rm** (audit logs)
- NodeManager: **/var/log/Bigdata/yarn/nm** (run logs) and **/var/log/Bigdata/audit/yarn/nm** (audit logs)

Log archive rule: The automatic compression and archive function is enabled for Yarn logs. By default, when the size of a log file exceeds 50 MB, the log file is automatically compressed. The naming rule of the compressed log file is as follows: *<Original log file name>-<yyyy-mm-dd_hh-mm-ss>.[ID]***.log.zip**. A maximum of 100 latest compressed files are retained. The number of compressed files can be configured on Manager.

Log archive rule:

Table 12-467 Yarn log list

Log Type	Log File Name	Description
Run log	hadoop-<SSH_USER>-<process_name>-<hostname>.log	Yarn component log file that records most of the logs generated when the Yarn component is running
	hadoop-<SSH_USER>-<process_name>-<hostname>.out	Log file that records Yarn running environment information
	<process_name>-<SSH_USER>-<DATE>-<PID>-gc.log	Garbage collection log file

Log Type	Log File Name	Description
	yarn-haCheck.log	ResourceManager active/standby status detection log file
	yarn-service-check.log	Log file that records the health check details of the Yarn service
	yarn-start-stop.log	Log file that records the startup and stop of the Yarn service
	yarn-prestart.log	Log file that records cluster operations before the Yarn service startup
	yarn-postinstall.log	Work log file after installation and before startup of the Yarn service
	hadoop-commission.log	Yarn service entry log file
	yarn-cleanup.log	Log file that records the cleanup operation during uninstallation of the Yarn service
	yarn-refreshqueue.log	Yarn queue refresh log file
	upgradeDetail.log	Upgrade log file
	stderr/stdin/syslog	Container log file of the applications running on the Yarn service
	yarn-application-check.log	Check log file of applications running on the Yarn service
	yarn-appsummary.log	Running result log file of applications running on the Yarn service
	yarn-switch-resourcemanager.log	Run log file that records the Yarn active/standby switchover
	ranger-yarn-plugin-enable.log	Log file that records the enabling of Ranger authentication for Yarn
	yarn-nodemanager-period-check.log	Periodic check log of Yarn NodeManager

Log Type	Log File Name	Description
	yarn-resourcemanager-period-check.log	Periodic check log of Yarn ResourceManager
	hadoop.log	Hadoop client logs
	env.log	Environment information log file before the instance is started or stopped.
Audit logs	yarn-audit-<process_name>.log ranger-plugin-audit.log	Yarn operation audit log file
	SecurityAuth.audit	Yarn security audit log file

Log Level

Table 12-468 describes the log levels supported by Yarn, including OFF, FATAL, ERROR, WARN, INFO, and DEBUG, from high priority to low. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-468 Log levels

Level	Description
FATAL	Logs of this level record critical error information about the current event processing.
ERROR	Logs of this level record error information about the current event processing.
WARN	Logs of this level record exception information about the current event processing.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system as well as system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of the Yarn service by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Click **Save Configuration**. In the dialog box that is displayed, click **OK** to make the setting take effect.

NOTE

The configurations take effect immediately without the need to restart the service.

----End

Log Format

The following table lists the Yarn log formats.

Table 12-469 Log formats

Log Type	Format	Example
Run log	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2014-09-26 14:18:59,109 | INFO | main | Client environment:java.compiler=<NA> | org.apache.zookeeper.Environment.logEnv(Environment.java:100)
Audit log	<yyyy-MM-dd HH:mm:ss,SSS>|<Log Level>|<*Thread that generates the log*>|<*Message in the log*>|<*Location of the log event*>	2014-09-26 14:24:43,605 | INFO | main-EventThread | USER=omm OPERATION=refreshAdminAcls TARGET=AdminService RESULT=SUCCESS | org.apache.hadoop.yarn.server.resourcemanager.RMAuditLogger$LogLevel$6.printLog(RMAuditLogger.java:91)

12.27.19 Yarn Performance Tuning

12.27.19.1 Preempting a Task

Scenario

The capacity scheduler of ResourceManager implements job preemption to simplify job running in queues and improve resource utilization. The process is as follows:

1. Assume that there are two queues (Queue A and Queue B). The capacity of Queue A is 25%, and the capacity of Queue B is 75%.
2. In the initial state, Task 1 is distributed to Queue A for processing, requiring 75% cluster resources. Task 2 is distributed to Queue B for processing, requiring 50% cluster resources.

3. Task 1 uses 25% cluster resources provided by Queue A and 50% resources from Queue B. Queue B reserves 25% cluster resources.
4. If task preemption is enabled, the resources of Task 1 will be preempted. Queue B preempts 25% cluster resources from Queue A for Task 2.
5. Task 1 will be executed when Task 2 is complete and the cluster has sufficient resources.

Procedure

Navigation path for setting parameters:

Go to the **All Configurations** page of Yarn and enter a parameter name in the search box by referring to **Modifying Cluster Service Configuration Parameters**.

Table 12-470 Parameter description

Parameter	Description	Default Value
yarn.resourcemanager.scheduler.monitor.enable	Whether to start scheduler monitoring according to **yarn.resourcemanager.scheduler.monitor.policies**. If this parameter is set to **true**, scheduler monitoring is enabled based on policies specified by **yarn.resourcemanager.scheduler.monitor.policies** and task resource preemption is enabled based on the scheduler information. If this parameter is set to **false**, scheduler monitoring is disabled.	false
yarn.resourcemanager.scheduler.monitor.policies	List of the SchedulingEditPolicy class to be used with the scheduler	org.apache.hadoop.yarn.server.resourcemanager.monitor.capacity.ProportionalCapacityPreemptionPolicy
yarn.resourcemanager.monitor.capacity.preemption.observe_only	• If this parameter is set to **true**, policies will be applied but task resource preemption will not be performed. • If this parameter is set to **false**, policies will be applied and task resource preemption will be performed based on the policies.	false

Parameter	Description	Default Value
yarn.resourcemanager.monitor.capacity.preemption.monitoring_interval	Monitoring interval, in millisecond. If this parameter is set to a larger value, capacity detection will not be performed frequently.	3000
yarn.resourcemanager.monitor.capacity.preemption.max_wait_before_kill	Interval between the time when a resource preemption request is sent and the time when the container is stopped (resources are released), in millisecond. The value must be greater than or equal to **0**. By default, if ApplicationMaster does not stop the container within 15 seconds, ResourceManager will forcibly stop the container after 15 seconds.	15000
yarn.resourcemanager.monitor.capacity.preemption.total_preemption_per_round	Maximum resource preemption ratio in a period. This value can be used to limit the speed at which containers are reclaimed from the cluster. After the expected total preemption value is calculated, the policy scales the preemption ratio back to this limit.	0.1
yarn.resourcemanager.monitor.capacity.preemption.max_ignored_over_capacity	Resource preemption dead zone = Total number of resources in the cluster x Value of this configuration item + Original resources of a queue (for example, Queue A). When resources actually used by a task in Queue A exceeds the preemption dead zone, the resource beyond the preemption dead zone is preempted. The value range is 0 to 1. **NOTE** A smaller value is recommended for effective preemption.	0

Parameter	Description	Default Value
yarn.resourcemanager.monitor.capacity.preemption.natural_termination_factor	Preemption percentage. Containers preempt only this percentage of the resources. For example, a termination factor of 0.5 will reclaim almost 95% of resources within 5 times of **yarn.resourcemanager.monitor.capacity.preemption.max_wait_before_kill**, even in the absence of natural termination. That is, 5 consecutive preemptions will be performed and each time half of the target resources will be preempted. The trend is geometric convergence. The interval of each preemption is **yarn.resourcemanager.monitor.capacity.preemption.max_wait_before_kill**. The value range is 0 to 1.	1

12.27.19.2 Setting the Task Priority

Scenario

The resource contention scenarios of a cluster are as follows:

1. Submit two jobs (Job 1 and Job 2) with lower priorities.
2. Some tasks of running Job 1 and Job 2 are in the running state. However, some tasks are pending due to resource deficiency because the capacity of cluster or queue resources is limited.
3. Submit a job (Job 3) with a higher priority. In this case, after the running tasks of Job 1 and Job 2 are complete, their resources will be released and then allocated to the pending tasks of Job 3.
4. After Job 3 is complete, its resources will be released and then allocated to Job 1 and Job 2.

Users can use capacity scheduler of ResourceManager to set the task priority in Yarn because the task priority is implemented by the scheduler of ResourceManager.

Procedure

Set the **mapreduce.job.priority** parameter and use CLI or API to set the task priority.

- Through the CLI

  When submitting tasks, add the **-Dmapreduce.job.priority=***<priority>* parameter.

<priority> can be set to any of the following values:

- VERY_HIGH
- HIGH
- NORMAL
- LOW
- VERY_LOW

- Through the API

  You can also set the task priority through the API.

  Set **Configuration.set("mapreduce.job.priority", <priority>)** or **Job.setPriority(JobPriority priority)**.

12.27.19.3 Optimizing Node Configuration

Scenario

After the scheduler of a big data cluster is properly configured, you can adjust the available memory, CPU resources, and local disk of each node to optimize the performance.

The configuration items are as follows:

- Available memory
- Number of vCPUs
- Physical CPU usage
- Coordination of memory and CPU resources
- Local disk

Procedure

For details about how to adjust parameter settings, see **Modifying Cluster Service Configuration Parameters**.

- **Available memory**

  Except the memory allocated to the OS and other services, allocate as much as possible memory to Yarn. You can adjust the following parameters to improve resource utilization.

  Assume that a container uses 512 MB memory by default, then the memory usage formula is: 512 MB x Number of containers.

  By default, the Map or Reduce container uses one vCPU and 1,024 MB memory, and ApplicationMaster uses 1,536 MB memory.

Parameter	Description	Default Value
yarn.nodemanager.resource.memory-mb	Physical memory that can be allocated to containers, in MB. The value must be greater than 0. You are advised to set the parameter value to 75% to 90% of the total physical memory of nodes. If the node has permanent processes of other services, reduce this parameter value to reserve sufficient resources for the processes.	MRS 3.*x* or later: **16384** Versions earlier than MRS 3.x: **8192**

- **Number of vCPUs**

  You are advised to set this parameter to 1.5 to 2 times the number of logical CPUs. If the upper layer computing applications have low computing capability requirements, you can set the parameter to two times the number of logical CPUs.

Parameter	Description	Default Value
yarn.nodemanager.resource.cpu-vcores	Number of vCPUs that can be used by Yarn on the node. The default value is **8**. You are advised to set the value to 1.5 to 2 times the number of logical CPUs.	8

- **Physical CPU usage**

  You are advised to reserve appropriate CPUs for the OS and the processes, such as database and HBase, and allocate the remaining CPUs to Yarn. You can set the following parameters to adjust the physical CPU usage.

Parameter	Description	Default Value
yarn.nodemanager.resource.percentage-physical-cpu-limit	Physical CPU percentage that can be used by Yarn on a node. The default value is **90**, indicating that no CPU control is implemented and Yarn can use all CPU resources. You can only view the parameter. To change the value of this parameter, set the value of RES_CPUSET_PERCENTAGE of YARN. You are advised to set this parameter to the percentage of CPU resources that can be used by the YARN cluster. For example, If 20% of CPU resources are used by other services (such as HBase, HDFS, and Hive) and system processes on the node, the CPU resources can be scheduled for Yarn is 1 - 20% = 80%. Therefore, you can set this parameter to **80**.	90

- **Local disk**

  MapReduce writes the intermediate job execution results in local disks. Therefore, configure disks as much as possible and disk space as large as possible. A simple way is to configure the same number of disks as DataNode except for the last directory.

  NOTE

  Use commas (,) to separate multiple disks.

Parameter	Description	Default Value
yarn.nodemanager.log-dirs	Directories in which logs are stored. Multiple directories can be specified. Storage location of container logs. The default value is **%{@auto.detect.datapart.nm.logs}**. If there is a data partition, a path list similar to **/srv/BigData/hadoop/data1/nm/containerlogs,/srv/BigData/hadoop/data2/nm/containerlogs** is generated based on the data partition. If there is no data partition, the default path **/srv/BigData/yarn/data1/nm/containerlogs** is generated. In addition to using expressions, you can enter a complete list of paths, such as **/srv/BigData/yarn/data1/nm/containerlogs** or **/srv/BigData/yarn/data1/nm/containerlogs,/srv/BigData/yarn/data2/nm/containerlogs**. In this way, data is stored in all the configured directories, which are usually on different devices. To ensure disk I/O load balancing, you are advised to provide several paths and each path corresponds to an independent disk. The localized log directory of the application exists in the relative path **/application_%{appid}**. The log directory of an independent container, that is, **container_{$contid}**, is the subdirectory of this directory. Each container directory contains the **stderr**, **stdin**, and **syslog** files generated by the container. To add a directory, for example, **/srv/BigData/yarn/data2/nm/containerlogs**, you need to delete the files in **/srv/BigData/yarn/data2/nm/containerlogs** first. Then, assign the same read and write permissions to **/srv/BigData/yarn/data2/nm/containerlogs** as those of **/srv/**	%{@auto.detect.datapart.nm.logs}

Parameter	Description	Default Value
	BigData/yarn/data1/nm/containerlogs, and change **/srv/BigData/yarn/data1/nm/containerlogs** to **/srv/BigData/yarn/data1/nm/containerlogs,/srv/BigData/yarn/data2/nm/containerlogs**. You can add directories, but do not modify or delete existing directories. Otherwise, NodeManager data will be lost and services will be unavailable. Default value: **%{@auto.detect.datapart.nm.logs}** Exercise caution when modifying this parameter. If the configuration is incorrect, the services are unavailable. If the value of this configuration item at the role level is changed, the value of this configuration item at all instance levels will be changed. If the value of this configuration item at the instance level is changed, the value of this configuration item of other instances remains unchanged.	

Parameter	Description	Default Value
yarn.nodemanager.local-dirs	Storage location of files after localization. The default value is **%{@auto.detect.datapart.nm.localdir}**. If there is a data partition, a path list similar to **/srv/BigData/hadoop/data1/nm/localdir,/srv/BigData/hadoop/data2/nm/localdir** is generated based on the data partition. If there is no data partition, the default path **/srv/BigData/yarn/data1/nm/localdir** is generated. In addition to using expressions, you can enter a complete list of paths, such as **/srv/BigData/yarn/data1/nm/localdir** or **/srv/BigData/yarn/data1/nm/localdir,/srv/BigData/yarn/data2/nm/localdir**. In this way, data is stored in all the configured directories, which are usually on different devices. To ensure disk I/O load balancing, you are advised to provide several paths and each path corresponds to an independent disk. The localized file directory of the application is stored in the relative path **/usercache/%{user}/appcache/application_%{appid}**. The working directory of an independent container, that is, **container_%{contid}**, is the subdirectory of the directory. To add a directory, for example, **/srv/BigData/yarn/data2/nm/localdir**, you need to delete the files in **/srv/BigData/yarn/data2/nm/localdir** first. Then, assign the same read and write permissions to **/srv/BigData/hadoop/data2/nm/localdir** as those of **/srv/BigData/hadoop/data1/nm/localdir**, and change **/srv/BigData/yarn/data1/nm/localdir** to **/srv/BigData/yarn/data1/nm/localdir,/srv/BigData/yarn/data2/nm/localdir**. You can add	%{@auto.detect.datapart.nm.localdir}

Parameter	Description	Default Value
	directories, but do not modify or delete existing directories. Otherwise, NodeManager data will be lost and services will be unavailable. Default value: **%{@auto.detect.datapart.nm.localdir}** Exercise caution when modifying this parameter. If the configuration is incorrect, the services are unavailable. If the value of this configuration item at the role level is changed, the value of this configuration item at all instance levels will be changed. If the value of this configuration item at the instance level is changed, the value of this configuration item of other instances remains unchanged.	

12.27.20 Common Issues About Yarn

12.27.20.1 Why Mounted Directory for Container is Not Cleared After the Completion of the Job While Using CGroups?

Question

Why mounted directory for Container is not cleared after the completion of the job while using CGroups?

Answer

The mounted path for the Container should be cleared even if job is failed.

This happens due to the deletion timeout. Some task takes more time to complete than the deletion time.

To avoid this scenario, you can go to the **All Configurations** page of Yarn by referring to **Modifying Cluster Service Configuration Parameters**. Search for the **yarn.nodemanager.linux-container-executor.cgroups.delete-timeout-ms** configuration item in the search box to change the deletion interval. The value is in milliseconds.

12.27.20.2 Why the Job Fails with HDFS_DELEGATION_TOKEN Expired Exception?

Question

Why is the HDFS_DELEGATION_TOKEN expired exception reported when a job fails in security mode?

Answer

HDFS_DELEGATION_TOKEN expires because the token is not updated or it is accessed after max. lifetime.

Ensure the following parameter value of max. lifetime of the token is greater than the job running time.

dfs.namenode.delegation.token.max-lifetime=**604800000** (1 week by default)

Go to the **All Configurations** page of HDFS by referring to **Modifying Cluster Service Configuration Parameters** and search for this parameter in the search box.

NOTE

You are advised to set this parameter to a value that is multiple times of the number of hours within the max. lifecycle of the token.

12.27.20.3 Why Are Local Logs Not Deleted After YARN Is Restarted?

Question

If Yarn is restarted in either of the following scenarios, local logs will not be deleted as scheduled and will be retained permanently:

- When Yarn is restarted during task running, local logs are not deleted.
- When the task is complete and logs fail to be collected, restart Yarn before the logs are cleared as scheduled. In this case, local logs are not deleted.

Answer

NodeManager has a restart recovery mechanism (for details, see **https://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-site/NodeManager.html#NodeManager_Restart**). Go to the **All Configurations** page of Yarn by referring to **Modifying Cluster Service Configuration Parameters**. Set **yarn.nodemanager.recovery.enabled** of NodeManager to **true** to make the configuration take effect. The default value is **true**. In this way, redundant local logs are periodically deleted when the YARN is restarted.

12.27.20.4 Why the Task Does Not Fail Even Though AppAttempts Restarts for More Than Two Times?

Question

Why the task does not fail even though AppAttempts restarts due to failure for more than two times?

Answer

During the task execution process, if the **ContainerExitStatus** returns value **ABORTED**, **PREEMPTED**, **DISKS_FAILED**, or **KILLED_BY_RESOURCEMANAGER**, the system will not count it as a failed attempt. Therefore, the task fails only when the AppAttempts fails actually, that is, the return value is not **ABORTED**, **PREEMPTED**, **DISKS_FAILED**, or **KILLED_BY_RESOURCEMANAGER** for two times.

12.27.20.5 Why Is an Application Moved Back to the Original Queue After ResourceManager Restarts?

Question

After I moved an application from one queue to another, why is it moved back to the original queue after ResourceManager restarts?

Answer

This problem is caused by the constraints of the ResourceManager. If a running application is moved to another queue, information about the new queue will not be stored in the ResourceManager after the ResourceManager restarts.

Assume that a user submits a MapReduce application to the leaf queue test11. If the leaf queue test11 is deleted when the application is running, the application will go to the lost_and found queue and the application stops. To start the application, the user moves the application to the leaf queue test21 and the application resumes running. If the ResourceManager restarts, the displayed submission queue is lost_and_found, but not test21.

If the application is not complete, the ResourceManager only stores the queue information before the application is moved. As a result, the application is moved back to the original queue. To solve this problem, move the application again after the ResourceManager is restarted to write information about the new queue to the ResourceManager.

12.27.20.6 Why Does Yarn Not Release the Blacklist Even All Nodes Are Added to the Blacklist?

Question

Why does Yarn not release the blacklist even all nodes are added to the blacklist?

Answer

In Yarn, when the number of application nodes added to the blacklist by ApplicationMaster (AM) reaches a certain proportion (the default value is 33% of the total number of nodes), the AM automatically releases the blacklist. In this way, all available nodes are added to the blacklist and tasks can obtain node resources.

Assume that there are 8 nodes in a cluster and they are divided in to pool A and pool B by NodeLabel. There are two nodes in pool B. A user submits a task App1 to pool B, but there is not sufficient HDFS space and App1 fails to run. As a result, two nodes in pool B are added to the blacklist by the AM of App1. According to the preceding principles, 2 is less than the 33% of 8. Therefore, Yarn does not release the blacklist, and App1 cannot obtain resources and keeps running. Even if the node that is added to the blacklisted is recovered, App1 still cannot obtain resources.

The preceding principles do not apply to the resource pool scenario. Therefore, you can change the value of the client parameter **yarn.resourcemanager.am-scheduling.node-blacklisting-disable-threshold** to **(nodes number of pool / total nodes)* 33%** to solve this problem.

12.27.20.7 Why Does the Switchover of ResourceManager Occur Continuously?

Question

The switchover of ResourceManager occurs continuously when multiple, for example 2,000, tasks are running concurrently, causing the Yarn service unavailable.

Answer

The cause is that the time of full GabageCollection exceeds the interaction duration threshold between the ResourceManager and ZooKeeper duration threshold. As a result, the connection between the ResourceManager and ZooKeeper fails and the switchover of ResourceManager occurs continuously.

When there are multiple tasks, ResourceManager saves the authentication information about multiple tasks and transfers the information to NodeManagers through heartbeat, which is called heartbeat response. The lifecycle of heartbeat response is short. The default value is 1s. Normally, heartbeat response can be reclaimed during the JVM minor GabageCollection. However, if there are multiple tasks and there are a lot of nodes, for example 5000 nodes, in the cluster, the heartbeat response of multiple nodes occupy a large amount of memory. As a result, the JVM cannot completely reclaim the heartbeat response during minor GabageCollection. The heartbeat response failed to be reclaimed accumulate and the JVM full GabageCollection is triggered. The JVM GabageCollection is in a blocking mode, in other words, no jobs are performed during the GabageCollection. Therefore, if the duration of full GabageCollection exceeds the periodical interaction duration threshold between the ResourceManager and ZooKeeper, the switchover occurs.

Log in to FusionInsight Manager, choose **Cluster** > **Services** > **Yarn**, and click the **Configurations** tab and then **All Configurations**. In the navigation pane on the

left, choose **Yarn** > **Customization**, and add the **yarn.resourcemanager.zk-timeout-ms** parameter to the **yarn.yarn-site.customized.configs** file to increase the threshold of the periodic interaction duration between ResourceManager and ZooKeeper (the value range is less than or equal to 90,000 ms). In this way, the problem of continuous active/standby ResourceManager switchover can be solved.

12.27.20.8 Why Does a New Application Fail If a NodeManager Has Been in Unhealthy Status for 10 Minutes?

Question

Why does a new application fail if a NodeManager has been in unhealthy status for 10 minutes?

Answer

When **nodeSelectPolicy** is set to **SEQUENCE** and the first NodeManager connected to the ResourceManager is unavailable, the ResourceManager attempts to assign tasks to the same NodeManager in the period specified by **yarn.nm.liveness-monitor.expiry-interval-ms**.

You can use either of the following methods to avoid the preceding problem:

- Use another nodeSelectPolicy, for example, **RANDOM**.
- Go to the **All Configurations** page of Yarn by referring to **Modifying Cluster Service Configuration Parameters**. Search for the following parameters in the search box and modify the following attributes in the **yarn-site.xml** file:

  yarn.resourcemanager.am-scheduling.node-blacklisting-enabled = **true**;

  yarn.resourcemanager.am-scheduling.node-blacklisting-disable-threshold = **0.5**.

12.27.20.9 Why Does an Error Occur When I Query the ApplicationID of a Completed or Non-existing Application Using the RESTful APIs?

Question

Why does an error occur when I query the applicationID of a completed or non-existing application using the RESTful APIs?

Answer

The Superior scheduler only stores the applicationIDs of running applications. If you view the applicationID of a completed or non-existing application by accessing the RESTful API at **https://<SS_REST_SERVER>/ws/v1/sscheduler/applications/{application_id}**. the 404 error is returned by the server. If Chrome web browser is used, the **Error Occurred** message is displayed because chrome preferentially responds in the application/xml format. If Internet Explorer is used, the **404** error code is displayed because IE web browser preferentially responds in the application/json format.

12.27.20.10 Why May A Single NodeManager Fault Cause MapReduce Task Failures in the Superior Scheduling Mode?

Question

In Superior scheduling mode, if a single NodeManager is faulty, why may the MapReduce tasks fail?

Answer

In normal cases, when the attempt of a single task of an application fails on a node for three consecutive times, the AppMaster of the application adds the node to the blacklist. Then, the AppMaster instructs the scheduler not to schedule the task to the node to avoid task failure.

However, by default, if 33% nodes in the cluster are added to the blacklist, the scheduler ignores the blacklisted nodes. Therefore, the blacklist feature is prone to become invalid in small cluster scenarios. For example, there are only three nodes in the cluster. If one node is faulty, the blacklist mechanism becomes invalid. The scheduler continues to schedule the task to the node no matter how many times the attempt of the task fails on the node. As a result, the number of attempts of the task reaches the maximum (4 times by default for MapReduce). And the MapReduce tasks failed.

Workaround:

The **yarn.resourcemanager.am-scheduling.node-blacklisting-disable-threshold** parameter indicates the threshold for ignoring blacklisted nodes, in percentage. You are advised to increase the value of this parameter based on the cluster scale. For example, you are advised to set this parameter to **50%** for a three-node cluster.

NOTE

The framework design of the Superior scheduler is time-based asynchronous scheduling. When the NodeManager is faulty, ResourceManager cannot quickly detect that the NodeManager is faulty (10 minutes by default). Therefore, the Superior scheduler still schedules tasks to the node, causing task failures.

12.27.20.11 Why Are Applications Suspended After They Are Moved From Lost_and_Found Queue to Another Queue?

Question

When a queue is deleted when there are applications running in it, these applications are moved to the "lost_and_found" queue. When these applications are moved back to another healthy queue, some tasks are suspended.

Answer

If no label expression is set for the current application, the default label expression of the queue is used as label expression for new container/resource demands requested by the application. If there is no default label expression of the queue,

then **default label** is considered as the label expression for new container/ resource demands requested by the application.

When application app1 is submitted to the queue Q1, **label1**, the default label expression of the queue, is used for the application's new resource requests/ containers. If Q1 is deleted when app1 is running, app1 is moved to the "lost_and_found" queue. Because there is no label expression of the "lost_and_found" queue, **default label** is used as the label expression of app1's new resource requests/containers. Assume that app1 is moved to another normal queue Q2. If Q2 supports **label1** and **default label**, app1 can run properly. If Q2 does not support **label1** or **default label**, the resource request with **label1** or **default label** cannot obtain resources, causing task suspension.

To solve this problem, ensure that the queue to which the application is moved from "lost_and_found" queue supports label expression of the moved application.

You are not advised to delete a queue in which there are running applications.

12.27.20.12 How Do I Limit the Size of Application Diagnostic Messages Stored in the ZKstore?

Question

How do I limit the size of application diagnostic messages stored in the ZKstore?

Answer

In some cases, it has been observed that diagnostic messages may grow infinitely. Because diagnostic messages are stored in the ZKstore, it is not recommended that you allow diagnostic messages to grow indefinitely. Therefore, a property parameter is needed to set the maximum size of the diagnostic message.

If you need to set **yarn.app.attempt.diagnostics.limit.kc**, go to the **All Configurations** page by referring to **Modifying Cluster Service Configuration Parameters** and search for the following parameters in the search box:

Table 12-471 Parameter description

Parameter	Description	Default Value
yarn.app.attempt.diagnostics.limit.kc	Data size of the diagnosis message for each application connection, in kilobytes (number of characters x 1,024). When ZooKeeper is used to store the behavior status of applications, the size of diagnosis messages needs to be limited to prevent Yarn from overloading ZooKeeper. If **yarn.resourcemanager.state-store.max-completed-applications** is set to a large value, you need to decrease the value of this property to limit the total size of stored data.	64

12.27.20.13 Why Does a MapReduce Job Fail to Run When a Non-ViewFS File System Is Configured as ViewFS?

Question

Why does a MapReduce job fail to run when a non-ViewFS file system is configured as ViewFS?

Answer

When a non-ViewFS file system is configured as a ViewFS using cluster, the user permissions on folders in the ViewFS file system are different from those of non-ViewFS folders in the default NameService. The submitted MapReduce job fails to be executed because the directory permissions are inconsistent.

When configuring the ViewFS user in the cluster, you need to check and verify the directory permissions. Before submitting a job, change the ViewFS folder permissions based on the default NameService folder permissions.

The following table lists the default permission structure of directories configured in ViewFS. If the configured directory permissions are not included in the following table, you must change the directory permissions accordingly.

Table 12-472 Default permission structure of directories configured in ViewFS

Parameter	Description	Default Value	Default value and default permissions on the parent directory
yarn.nodemanager.remote-app-log-dir	On the default file system (usually HDFS), specify the directory to which the NM aggregates logs.	logs	777
yarn.nodemanager.remote-app-log-archive-dir	Directory for archiving logs	-	777
yarn.app.mapreduce.am.staging-dir	Staging directory used when a job is submitted	/tmp/hadoop-yarn/staging	777
mapreduce.jobhistory.intermediate-done-dir	Directory for storing historical files of MapReduce jobs	${yarn.app.mapreduce.am.staging-dir}/history/done_intermediate	777

Parameter	Description	Default Value	Default value and default permissions on the parent directory
mapreduce.jobhistory.done-dir	Directory of historical files managed by the MR JobHistory Server.	${yarn.app.mapreduce.am.staging-dir}/history/done	777

12.27.20.14 Why Do Reduce Tasks Fail to Run in Some OSs After the Native Task Feature is Enabled?

Question

After the Native Task feature is enabled, Reduce tasks fail to run in some OSs.

Answer

When **-Dmapreduce.job.map.output.collector.class=org.apache.hadoop.mapred.nativetask.NativeMapOutputCollectorDelegator** is executed to enable the Native Task feature during the running of MapReduce tasks that contain Reduce tasks, the tasks fail to run in some OSs, and the error message "version 'GLIBCXX_3.4.20' not found" is displayed in logs. The cause is that the GLIBCXX version of the OSs is too early. As a result, the libnativetask.so.1.0.0 library on which the feature depends cannot be loaded, leading to task failures.

Workaround:

Set **mapreduce.job.map.output.collector.class** to **org.apache.hadoop.mapred.MapTask$MapOutputBuffer**.

12.28 Using ZooKeeper

12.28.1 Using ZooKeeper from Scratch

ZooKeeper is an open-source, highly reliable, and distributed consistency coordination service. ZooKeeper is designed to solve the problem that data consistency cannot be ensured for complex and error-prone distributed systems. There is no need to develop dedicated collaborative applications, which is suitable for high availability services to ensure data consistency.

Background Information

Before using the client, you need to download and update the client configuration file on all clients except the client of the active management node.

Procedure

For MRS 2.*x* or earlier, perform the following operations:

Step 1 Download the client configuration file.

1. Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click the cluster to be operated. This cluster is the one created in .
2. Click the **Components** tab.
3. Click **Services** and then **Download Client**.

   Set **Client Type** to **Only configuration files**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default. You can customize the file path.

Step 2 Log in to the active management node of MRS Manager.

1. On the MRS console, choose **Clusters > Active Clusters** and click a cluster name. On the **Nodes** tab, view the node names. The node whose name contains **master1** is the Master1 node, and the node whose name contains **master2** is the Master2 node.

   The active and standby management nodes of MRS Manager are installed on Master nodes by default. Because Master1 and Master2 are switched over in active and standby mode, Master1 is not always the active management node of MRS Manager. Run a command in Master1 to check whether Master1 is active management node of MRS Manager. For details about the command, see **Step 2.4**.
2. Log in to the Master1 node using the password as user **root**. For details, see **Help Center** > **MapReduce Service** > **Connecting to Clusters** > **Logging In to a Cluster** > **Logging In to an ECS**.
3. Run the following commands to switch to user **omm**:

   sudo su - root

   su - omm
4. Run the following command to check the active management node of MRS Manager:

   sh ${BIGDATA_HOME}/om-0.0.1/sbin/status-oms.sh

   In the command output, the node whose **HAActive** is **active** is the active management node, and the node whose **HAActive** is **standby** is the standby management node. In the following example, **mgtomsdat-sh-3-01-1** is the active management node, and **mgtomsdat-sh-3-01-2** is the standby management node.

   ```
   Ha mode
   double
   NodeName            HostName                HAVersion       StartTime           HAActive
   HAAllResOK          HARunPhase
   192-168-0-30        mgtomsdat-sh-3-01-1     V100R001C01     2014-11-18 23:43:02
   active              normal                  Actived
   192-168-0-24        mgtomsdat-sh-3-01-2     V100R001C01     2014-11-21 07:14:02
   standby             normal                  Deactived
   ```
5. Log in to the active management node, for example, **192-168-0-30** of MRS Manager as user **root**, and run the following command to switch to user **omm**:

sudo su - omm

Step 3 Run the following command to go to the client installation directory, for example, **/opt/client**.

cd /opt/client

Step 4 Run the following command to update the client configuration for the active management node.

sh refreshConfig.sh /opt/client *Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/client/tmp/MRS-client/MRS_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully:

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

NOTE

You can perform **Step 1** to **Step 4** by referring to Method 2 in **User Guide** > **Connecting to Clusters** > **Using an MRS Client** > **Updating a Client**.

Step 5 Use the client on a Master node.

1. On the active management node where the client is updated, for example, node **192-168-0-30**, run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. If Kerberos authentication is disabled for the current cluster, skip this step:

   kinit *MRS cluster user*

   Example: **kinit zookeeperuser**.

4. Run the following Zookeeper client command:

   zkCli.sh -server *<zookeeper installation node IP>***:<port>**

   Example: ***zkCli.sh -server node-master1DGhZ:2181***

Step 6 Run the ZooKeeper client command.

1. Create a ZNode.
   ```
   create /test
   ```
2. View ZNode information.
   ```
   ls /
   ```
3. Write data to the ZNode.
   ```
   set /test "zookeeper test"
   ```
4. View the data written to the ZNode.
   ```
   get /test
   ```
5. Delete the created ZNode.
   ```
   delete /test
   ```

----End

For MRS 3.x or later, perform the following operations:

Step 1 Download the client configuration file.

1. Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.
2. Choose **Cluster** > *Name of the desired cluster* > **Dashboard** > **More** > **Download Client**.
3. Download the cluster client.

   Set **Select Client Type** to **Configuration Files Only**, select a platform type, and click **OK** to generate the client configuration file which is then saved in the **/tmp/FusionInsight-Client/** directory on the active management node by default.

Step 2 Log in to the active management node of Manager.

1. Log in to any node where Manager is deployed as user **root**.
2. Run the following command to identify the active and standby nodes:

   sh ${BIGDATA_HOME}/om-server/om/sbin/status-oms.sh

   In the command output, the value of **HAActive** for the active management node is **active**, and that for the standby management node is **standby**. In the following example, **node-master1** is the active management node, and **node-master2** is the standby management node.

   ```
   HAMode
   double
   NodeName            HostName        HAVersion       StartTime           HAActive
   HAAllResOK          HARunPhase
   192-168-0-30        node-master1    V100R001C01     2020-05-01 23:43:02 active
   normal              Actived
   192-168-0-24        node-master2    V100R001C01     2020-05-01 07:14:02 standby
   normal              Deactived
   ```

3. Log in to the primary management node as user **root** and run the following command to switch to user **omm**:

   sudo su - omm

Step 3 Run the following command to go to the client installation directory, for example, **/opt/client**.

cd /opt/client

Step 4 Run the following command to update the client configuration for the active management node.

sh refreshConfig.sh /opt/client *Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/client /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully:

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

Step 5 Use the client on a Master node.

1. On the active management node where the client is updated, for example, node **192-168-0-30**, run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication has been enabled for the current cluster, run the following command to authenticate the current user. For details, see to configure roles with required permissions. For details about how to bind roles with users, see . If Kerberos authentication is disabled for the current cluster, skip this step:

   kinit *MRS cluster user*

   Example: **kinit zookeeperuser**.

4. Run the following Zookeeper client command:

   zkCli.sh -server <zookeeper installation node IP>:<port>

   Example: **zkCli.sh -server node-master1DGhZ:2181**

Step 6 Run the ZooKeeper client command.

1. Create a ZNode.
   ```
   create /test
   ```
2. View ZNode information.
   ```
   ls /
   ```
3. Write data to the ZNode.
   ```
   set /test "zookeeper test"
   ```
4. View the data written to the ZNode.
   ```
   get /test
   ```
5. Delete the created ZNode.
   ```
   delete /test
   ```

----End

12.28.2 Common ZooKeeper Parameters

Navigation path for setting parameters:

Go to the **All Configurations** page of ZooKeeper by referring to **Modifying Cluster Service Configuration Parameters**. Enter a parameter name in the search box.

Table 12-473 Parameters

Parameter	Description	Default Value
skipACL	Specifies whether to skip the permission check of the ZooKeeper node.	no

Parameter	Description	Default Value
maxClientCnxns	Specifies the maximum number of connections of ZooKeeper. It is recommended this parameter is set to a larger value in scenarios with a large number of connections.	2000
LOG_LEVEL	Specifies the log level. This parameter can be set to **DEBUG** during commissioning.	INFO
acl.compare.shortName	Specifies whether to perform ACL authentication only by principal username when the Znode ACL authentication type is SASL.	true
synclimit	Specifies the interval of synchronization between the follower and leader (unit: tick). If the leader does not respond within the specified time range, the connection cannot be established.	15
tickTime	Specifies the duration of a tick (in milliseconds). It is the basic time unit used by ZooKeeper, which defines heartbeat and timeout durations.	4000

NOTE

The ZooKeeper internal time is determined by **ticktime** and **synclimit**. To increase the ZooKeeper internal timeout interval, increase the timeout interval for the client to connect to ZooKeeper.

12.28.3 Using a ZooKeeper Client

Scenario

Use a ZooKeeper client in an O&M scenario or service scenario.

Prerequisites

You have installed the client. For example, the installation directory is **/opt/client**. The client directory in the following operations is only an example. Change it based on the actual installation directory onsite.

Procedure

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to authenticate the user: (skip this step in common mode):

kinit *Component service user*

Step 5 Run the following command to log in to the client tool:

zkCli.sh -server *service IP address of the node where the ZooKeeper role instance locates*:*client port*

----End

12.28.4 Configuring the ZooKeeper Permissions

Scenario

Configure znode permission of ZooKeeper.

ZooKeeper uses an access control list (ACL) to implement znode access control. The ZooKeeper client specifies a znode ACL, and the ZooKeeper server determines whether a client that requests for a znode has related operation permission according to the ACL. ACL configuration involves the following four operations:

- Check znode ACLs in ZooKeeper.
- Add znode ACLs to ZooKeeper.
- Modify znode ACLs in ZooKeeper.
- Delete znode ACLs from ZooKeeper.

  The ZooKeeper ACL permission is described as follows:

  ZooKeeper supports five types of permission, create, delete, read, write, and admin. ZooKeeper permission control is of a znode level. That is, the permission configuration for a parent znode is not inherited by its child znodes. The ZooKeeper znode default permission is **world:anyone: cdrwa**. That is, any user has all permissions.

NOTE

ACL has three parts:

The first part is the authentication type. For example, **world** indicates all authentication types and **sasl** indicates the kerberos authentication type.

The second part is the account. For example, anyone indicates any user.

The third part is permission. For example, **cdrwa** indicates all permissions.

In particular, because starting the client in common mode does not need authentication, ACL with **sasl** authentication type cannot be used in common mode. Authentications of **sasl** scheme in this document are performed in clusters that have the security mode enabled.

Table 12-474 Five types of ZooKeeper ACLs

Permission Description	Permission Name	Permission Details
Create permission	create(c)	Users with this permission can create child znodes in the current znode.
Delete permission	delete(d)	Users with this permission can delete the current znode.
Read permission	read(r)	Users with this permission can obtain data of the current znode and list all the child znodes of the current znode.
Write permission	write(w)	Users with this permission can write data to the current znode and its child znodes.
Administration permission	admin(a)	Users with this permission can set permission for the current znode.

Impact on the System

NOTICE

Modifying ZooKeeper ACLs is a critical operation. If znode permission is modified in ZooKeeper, other users may have no permission to access the znode and some system functions are abnormal. In 3.5.6 and later versions, users must have the read permission for the **getAcl** operation.

Prerequisites

- The ZooKeeper client has been installed. For example, the installation directory is **/opt/client**.
- You have obtained the password of the system administrator account.

Procedure

Start the ZooKeeper client.

Step 1 Log in to the server where the ZooKeeper client is installed as user **root**.

Step 2 Run the following command to go to the client installation directory:

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 If the cluster has the security mode enabled, run the following command for user authentication and enter the username and password (Any authorized user. **admin** is used as an example.):

kinit admin

Step 5 On the ZooKeeper client, run the following command to go to the ZooKeeper command-line interface (CLI):

sh zkCli.sh -server *ZooKeeper plane IP address of any instance*:*clientPort*

The default **clientPort** is **2181**.

Example: **sh zkCli.sh -server 192.168.0.151:2181**

Step 6 Run the **ls** command to view the znode list in ZooKeeper. For example, you can view the list of znodes in the root directory.

ls /

```
[zk: 192.168.0.151:2181(CONNECTED) 1] ls /
[hadoop-flag, hadoop-ha, test, test2, test3, test4, test5, test6, zookeeper]
```

View the ZooKeeper znode ACL.

Step 7 Start the ZooKeeper client.

Step 8 Run the **getAcl** command to view znodes. The following command can be used to view the created znode ACL named **test**:

getAcl */znode name*

```
[zk: 192.168.0.151:2181(CONNECTED) 2] getAcl /test
'world,'anyone
: cdrwa
```

Add a ZooKeeper znode ACL.

Step 9 Start the ZooKeeper client.

Step 10 View the old ACL information to check whether the current account has the permission to modify the znode ACL information (a permission). If no, use kinit to switch to a user that has the permission and restart the ZooKeeper client.

getAcl */znode name*

```
 [zk: 192.168.0.151:2181(CONNECTED) 3] getAcl /test
'world,'anyone
: cdrwa
```

Step 11 Run the **setAcl** command to add an ACL. The command for adding an ACL is as follows:

setAcl /test world:anyone:cdrwa,sasl: *username*@: *<system domain name>:ACL value*

For example, to add the ACL of the **admin** user to the test znode, run the following command:

setAcl /test world:anyone:cdrwa,sasl:admin@HADOOP.COM:cdrwa

NOTE

When adding a new ACL, reserve the existing ones. The new and old ACLs are separated by a comma. The newly added ACL has three parts:

The first part is the authentication type. For example, **sasl** indicates the kerberos authentication type.

The second part is the account. For example, **admin@HADOOP.COM** indicates user **admin**.

The third part is permission. For example, **cdrwa** indicates all permissions.

Step 12 After adding the ACL, run the **getAcl** command to check whether the permission is added successfully:

getAcl */znode name*

```
[zk: 192.168.0.151:2181(CONNECTED) 4] getAcl /test
'world,'anyone
: cdrwa
'sasl,'admin@<system domain name>
: cdrwa
```

Modify the ZooKeeper znode ACL.

Step 13 Start the ZooKeeper client.

Step 14 View the old ACL information to check whether the current account has the permission to modify the znode ACL information (a permission). If no, use kinit to switch to a user that has the permission and restart the ZooKeeper client.

getAcl */znode name*

```
[zk: 192.168.0.151:2181(CONNECTED) 5] getAcl /test
'world,'anyone
: cdrwa
'sasl,'admin@<system domain name>
: cdrwa
```

Step 15 Run the **setAcl** command to modify an ACL. The command for adding an ACL is as follows:

s**etAcl /test sasl:***Username@<System domain name>:ACL value*

For example, to reserve only **admin** user permission and delete **anyone** rw permission, run the following command:

setAcl /test sasl:admin@HADOOP.COM:cdrwa

Step 16 After modifying the ACL, run the **getAcl** command to check whether the permission is modified successfully:

getAcl */znode name*

```
[zk: 192.168.0.151:2181(CONNECTED) 6] getAcl /test
'sasl,'admin@<system domain name>
: cdrwa
```

Delete the ZooKeeper znode ACL.

Step 17 Start the ZooKeeper client.

Step 18 View the old ACL information to check whether the current account has the permission to modify the znode ACL information (a permission). If no, use kinit to switch to a user that has the permission and restart the ZooKeeper client.

getAcl */znode name*

```
[zk: 192.168.0.151:2181(CONNECTED) 5] getAcl /test
'world,'anyone
: rw
'sasl,'admin@<system domain name>
: cdrwa
```

Step 19 Run the **setAcl** command to add an ACL. The command for adding an ACL is as follows:

s**etAcl /test sasl:***Username@<System domain name>:ACL value*

For example, to reserve only **admin** user permission and delete **anyone** rw permission, run the following command:

setAcl /test sasl:admin@HADOOP.COM:cdrwa

Step 20 After modifying the ACL, run the **getAcl** command to check whether the permission is modified successfully:

getAcl */znode name*

```
[zk: 192.168.0.151:2181(CONNECTED) 6] getAcl /test
'sasl,'admin@<system domain name>
: cdrwa
```

----End

12.28.5 ZooKeeper Log Overview

Log Description

Log path: **/var/log/Bigdata/zookeeper/quorumpeer** (Run log), **/var/log/Bigdata/audit/zookeeper/quorumpeer** (Audit log)

Log archive rule: The automatic ZooKeeper log compression function is enabled. By default, when the size of logs exceeds 30 MB, logs are automatically compressed into a log file. A maximum of 20 compressed file can be reserved. The number of compressed files can be configured on Manager.

Table 12-475 ZooKeeper log list

Log Type	Log File Name	Description
Run logs	zookeeper-<SSH_USER>-<process_name>-<hostname>.log	ZooKeeper system log file, which records most of the logs generated when the ZooKeeper system is running.
	check-serviceDetail.log	Log that records whether the ZooKeeper service starts successfully.
	zookeeper-<SSH_USER>-<DATA>-<PID>-gc.log	ZooKeeper garbage collection log file
	instanceHealthDetail.log	Log that records the health check details of ZooKeeper instance
	zookeeper-omm-server-<hostname>.out	Log indicating that ZooKeeper unexpectedly quits
	zk-err-<zkpid>.log	ZooKeeper fatal error log
	java_pid<zkpid>.hprof	ZooKeeper memory overflow log
	funcDetail.log	ZooKeeper instance startup log
	zookeeper-period-check.log	Health check log of the ZooKeeper instance
	zookeeper-period-check-java.log	ZooKeeper quota monitoring period check log
Audit Log	zk-audit-quorumpeer.log	ZooKeeper operation audit log

Log levels

Table 12-476 describes the log levels supported by ZooKeeper. The priorities of log levels are FATAL, ERROR, WARN, INFO, and DEBUG in descending order. Logs whose levels are higher than or equal to the specified level are printed. The number of printed logs decreases as the specified log level increases.

Table 12-476 Log levels

Level	Description
FATAL	Logs of this level record fatal error information about the current event processing that may result in a system crash.
ERROR	Error information about the current event processing, which indicates that system running is abnormal.

Level	Description
WARN	Abnormal information about the current event processing. These abnormalities will not result in system faults.
INFO	Logs of this level record normal running status information about the system and events.
DEBUG	Logs of this level record the system information and system debugging information.

To modify log levels, perform the following operations:

Step 1 Go to the **All Configurations** page of the ZooKeeper service by referring to **Modifying Cluster Service Configuration Parameters**.

Step 2 On the menu bar on the left, select the log menu of the target role.

Step 3 Select a desired log level.

Step 4 Click **Save**. In the displayed dialog box, click **OK** to make the configuration take effect.

NOTE

The configurations take effect immediately without the need to restart the service.

----End

Log Format

The following table lists the ZooKeeper log formats.

Table 12-477 Log Format

Log Type	Component	Format	Example
Run logs	zookeeper quorumpeer	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2020-01-20 16:33:43,816 | INFO | main | Defaulting to majority quorums | org.apache.zookeeper.server.quorum.QuorumPeerConfig.parseProperties(QuorumPeerConfig.java:335)

Log Type	Component	Format	Example
Audit logs	zookeeper quorumpeer	<*yyyy-MM-dd HH:mm:ss,SSS*>|<*Log level*>|<*Name of the thread that generates the log*>|<*Message in the log*>|<*Location where the log event occurs*>	2020-01-20 16:33:54,313 | INFO | CommitProcessor:13 | session=0xd4b0679daea0000 ip=10.177.112.145 operation=create znode target=ZooKeeperServer znode=/zk-write-test-2 result=success | org.apache.zookeeper.ZKAuditLogger$LogLevel$5.printLog(ZKAuditLogger.java:70)

12.28.6 Common Issues About ZooKeeper

12.28.6.1 Why Do ZooKeeper Servers Fail to Start After Many znodes Are Created?

Question

After a large number of znodes are created, ZooKeeper servers in the ZooKeeper cluster become faulty and cannot be automatically recovered or restarted.

Logs of followers:

```
2016-06-23 08:00:18,763 | WARN  | QuorumPeer[myid=26](plain=/10.16.9.138:2181)(secure=disabled) |
Exception when following the leader |
org.apache.zookeeper.server.quorum.Follower.followLeader(Follower.java:93)
java.net.SocketTimeoutException: Read timed out
    at java.net.SocketInputStream.socketRead0(Native Method)
    at java.net.SocketInputStream.socketRead(SocketInputStream.java:116)
    at java.net.SocketInputStream.read(SocketInputStream.java:170)
    at java.net.SocketInputStream.read(SocketInputStream.java:141)
    at java.io.BufferedInputStream.fill(BufferedInputStream.java:246)
    at java.io.BufferedInputStream.read(BufferedInputStream.java:265)
    at java.io.DataInputStream.readInt(DataInputStream.java:387)
    at org.apache.jute.BinaryInputArchive.readInt(BinaryInputArchive.java:63)
    at org.apache.zookeeper.server.quorum.QuorumPacket.deserialize(QuorumPacket.java:83)
    at org.apache.jute.BinaryInputArchive.readRecord(BinaryInputArchive.java:99)
    at org.apache.zookeeper.server.quorum.Learner.readPacket(Learner.java:156)
    at org.apache.zookeeper.server.quorum.Learner.registerWithLeader(Learner.java:276)
    at org.apache.zookeeper.server.quorum.Follower.followLeader(Follower.java:75)
    at org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.java:1094)
2016-06-23 08:00:18,764 | INFO  | QuorumPeer[myid=26](plain=/10.16.9.138:2181)(secure=disabled) |
shutdown called | org.apache.zookeeper.server.quorum.Follower.shutdown(Follower.java:198)
java.lang.Exception: shutdown Follower
```

```
    at org.apache.zookeeper.server.quorum.Follower.shutdown(Follower.java:198)
    at org.apache.zookeeper.server.quorum.QuorumPeer.stopFollower(QuorumPeer.java:1141)
    at org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.java:1098)
```

Logs of the leader:

```
2016-06-23 07:30:57,481 | WARN  | QuorumPeer[myid=25](plain=/10.16.9.136:2181)(secure=disabled) |
Unexpected exception | org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.java:1108)
java.lang.InterruptedException: Timeout while waiting for epoch to be acked by quorum
    at org.apache.zookeeper.server.quorum.Leader.waitForEpochAck(Leader.java:1221)
    at org.apache.zookeeper.server.quorum.Leader.lead(Leader.java:487)
    at org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.java:1105)
2016-06-23 07:30:57,482 | INFO  | QuorumPeer[myid=25](plain=/10.16.9.136:2181)(secure=disabled) |
Shutdown called | org.apache.zookeeper.server.quorum.Leader.shutdown(Leader.java:623)
java.lang.Exception: shutdown Leader! reason: Forcing shutdown
    at org.apache.zookeeper.server.quorum.Leader.shutdown(Leader.java:623)
    at org.apache.zookeeper.server.quorum.QuorumPeer.stopLeader(QuorumPeer.java:1149)
    at org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.java:1110)
```

Answer

After a large number of znodes are created, a large volume of data needs to be synchronized between the follower and leader. If the data synchronization is not complete within the specified time, all ZooKeeper servers fail to start.

Go to the **All Configurations** page of the ZooKeeper service by referring to **Modifying Cluster Service Configuration Parameters**. To recover ZooKeeper servers, increase the values of **syncLimit** and **initLimit** in the ZooKeeper configuration file **zoo.cfg** until ZooKeeper servers are successfully started.

Table 12-478 Parameters

Parameter	Description	Default Value
syncLimit	Interval (unit: tick) at which data is synchronized between the follower and the leader. If the leader does not respond to the follower within the specified time, the connection between the leader and follower cannot be set up.	15
initLimit	Interval (unit: tick) within which the connection and synchronization between the follower and leader must be completed.	15

If ZooKeeper servers do not recover even after **initLimit** and **syncLimit** are set to **300** ticks, check that no other application is killing the ZooKeeper. For example, if the parameter value is **300** and the ticket duration is 2000 ms, the maximum synchronization duration is 600s (300 x 2000 ms).

There may exist the situation where an overwhelming amount of data is created in ZooKeeper and it takes long to synchronize data between the follower and the leader and to save data to the hard disk. This means that ZooKeeper needs to run for a long time. Ensure that no other monitoring application kills the ZooKeeper while ZooKeeper is running.

12.28.6.2 Why Does the ZooKeeper Server Display the java.io.IOException: Len Error Log?

Question

After a large number of znodes are created in a parent directory, the ZooKeeper client will fail to fetch all child nodes of this parent directory in a single request.

Logs of client:

```
2017-07-11 13:17:19,610 [myid:] - WARN  [New I/O worker #3:ClientCnxnSocketNetty
$ZKClientHandler@468] - Exception caught: [id: 0xb66cbb85, /10.18.97.97:49192 :>
10.18.97.97/10.18.97.97:2181] EXCEPTION: java.nio.channels.ClosedChannelException
java.nio.channels.ClosedChannelException
at org.jboss.netty.handler.ssl.SslHandler$6.run(SslHandler.java:1580)
at org.jboss.netty.channel.socket.ChannelRunnableWrapper.run(ChannelRunnableWrapper.java:40)
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.executeInIoThread(AbstractNioWorker.java:71)
at org.jboss.netty.channel.socket.nio.NioWorker.executeInIoThread(NioWorker.java:36)
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.executeInIoThread(AbstractNioWorker.java:57)
at org.jboss.netty.channel.socket.nio.NioWorker.executeInIoThread(NioWorker.java:36)
at org.jboss.netty.channel.socket.nio.AbstractNioChannelSink.execute(AbstractNioChannelSink.java:34)
at org.jboss.netty.handler.ssl.SslHandler.channelClosed(SslHandler.java:1566)
at org.jboss.netty.channel.Channels.fireChannelClosed(Channels.java:468
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.close(AbstractNioWorker.java:376)
at org.jboss.netty.channel.socket.nio.NioWorker.read(NioWorker.java:93)
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.process(AbstractNioWorker.java:109)
at org.jboss.netty.channel.socket.nio.AbstractNioSelector.run(AbstractNioSelector.java:312)
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.run(AbstractNioWorker.java:90)
at org.jboss.netty.channel.socket.nio.NioWorker.run(NioWorker.java:178)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
at java.lang.Thread.run(Thread.java:745)
```

Logs of leader:

```
2017-07-11 13:17:33,043 [myid:1] - WARN  [New I/O worker #7:NettyServerCnxn@445] - Closing
connection to /10.18.101.110:39856
java.io.IOException: Len error 45
at org.apache.zookeeper.server.NettyServerCnxn.receiveMessage(NettyServerCnxn.java:438)
at org.apache.zookeeper.server.NettyServerCnxnFactory
$CnxnChannelHandler.processMessage(NettyServerCnxnFactory.java:267)
at org.apache.zookeeper.server.NettyServerCnxnFactory
$CnxnChannelHandler.messageReceived(NettyServerCnxnFactory.java:187)
at org.jboss.netty.channel.SimpleChannelHandler.handleUpstream(SimpleChannelHandler.java:88)
at org.jboss.netty.channel.DefaultChannelPipeline.sendUpstream(DefaultChannelPipeline.java:564)
at org.jboss.netty.channel.DefaultChannelPipeline.sendUpstream(DefaultChannelPipeline.java:559)
at org.jboss.netty.channel.Channels.fireMessageReceived(Channels.java:268)
at org.jboss.netty.channel.Channels.fireMessageReceived(Channels.java:255)
at org.jboss.netty.channel.socket.nio.NioWorker.read(NioWorker.java:88)
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.process(AbstractNioWorker.java:109)
at org.jboss.netty.channel.socket.nio.AbstractNioSelector.run(AbstractNioSelector.java:312)
at org.jboss.netty.channel.socket.nio.AbstractNioWorker.run(AbstractNioWorker.java:90)
at org.jboss.netty.channel.socket.nio.NioWorker.run(NioWorker.java:178)
at org.jboss.netty.util.ThreadRenamingRunnable.run(ThreadRenamingRunnable.java:108)
at org.jboss.netty.util.internal.DeadLockProofWorker$1.run(DeadLockProofWorker.java:42)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
at java.lang.Thread.run(Thread.java:745)
```

Answer

After a large number of znodes are created in a single parent directory and the client tries to fetch all the child znodes in a single request, the server will fail to return because the results exceed the data size that can be stored in a znode.

To avoid this problem, set **jute.maxbuffer** to a larger value based on the client application.

jute.maxbuffer can only be set to a Java system property without the Zookeeper prefix. To set **jute.maxbuffer** to *X*, set **Djute.maxbuffer** to *X* when starting the ZooKeeper client or the service.

For example, set the parameter to 4 MB: **-Djute.maxbuffer=0x400000**.

Table 12-479 Parameters

Parameter	Description	Default Value
jute.maxbuffer	Specifies the maximum length of data that can be stored in znode. The unit is byte. Default value: 0xfffff, which is less than 1 MB. **NOTE** If this option is changed, the system property must be set on all servers and clients, otherwise problems will arise.	0xfffff

12.28.6.3 Why Four Letter Commands Don't Work With Linux netcat Command When Secure Netty Configurations Are Enabled at Zookeeper Server?

Question

Why four letter commands do not work with linux netcat command when secure netty configurations are enabled at Zookeeper server?

For example,

echo stat /netcat host port

Answer

Linux ***netcat*** command does not have option to communicate Zookeeper server securely, so it cannot support Zookeeper four letter commands when secure netty configurations are enabled.

To avoid this problem, user can use below Java API to execute four letter commands.

```
org.apache.zookeeper.client.FourLetterWordMain
```

For example,

```
String[] args = new String[]{host, port, "stat"};
org.apache.zookeeper.client.FourLetterWordMain.main(args);
```

NOTE

netcat command should be used only with non secure netty configuration.

12.28.6.4 How Do I Check Which ZooKeeper Instance Is a Leader?

Question

How to check whether the role of a ZooKeeper instance is a leader or follower.

Answer

Log in to Manager and choose **Cluster** > *Name of the desired cluster* > **Service** > **ZooKeeper** > **Instance**. On the displayed page, click the name of the quorumpeer instance. On the displayed instance details page, view the server status of the instance.

12.28.6.5 Why Cannot the Client Connect to ZooKeeper using the IBM JDK?

Question

When the IBM JDK is used, the client fails to connect to ZooKeeper.

Answer

The possible cause is that the **jaas.conf** file format of the IBM JDK is different from that of the common JDK.

If IBM JDK is used, use the following **jaas.conf** template. The **useKeytab** file path must start with **file://**, followed by an absolute path.

```
Client {
com.ibm.security.auth.module.Krb5LoginModule required
useKeytab="file://D:/install/HbaseClientSample/conf/user.keytab"
principal="hbaseuser1"
credsType="both";
};
```

12.28.6.6 What Should I Do When the ZooKeeper Client Fails to Refresh a TGT?

Question

The ZooKeeper client fails to refresh a TGT and therefore ZooKeeper cannot be accessed. The error message is as follows:

```
Login: Could not renew TGT due to problem running shell command: '***/kinit -R'; exception
was:org.apache.zookeeper.Shell$ExitCodeException: kinit: Ticket expired while renewing credentials
```

Answer

ZooKeeper uses the system command **kinit – R** to refresh a ticket. In the current version of MRS, the function of this command is canceled. If a long-term task needs to be executed, you are advised to implement the authentication function in keytab mode.

In the **jaas.conf** configuration file, set **useTicketCache** to **false**, **useKeyTab** to **true**, and specify the keytab path.

12.28.6.7 Why Is Message "Node does not exist" Displayed when A Large Number of Znodes Are Deleted Using the deleteall Command

Question

When the client connects to a non-leader instance, run the **deleteall** command to delete a large number of znodes, the error message "Node does not exist" is displayed, but run the **stat** command, the node status can be obtained.

Answer

The leader and follower data is not synchronized due to network problems or large data volume. To solve this problem, connect the client to the leader instance and delete the instance. To delete the leader node, view the IP address of the node where the leader resides by referring to **How Do I Check Which ZooKeeper Instance Is a Leader?**, run the **zkCli.sh -server leader** *node IP address* **2181** command to connect to the client, and then run the **deleteall** command to delete the leader node. For details, see **Using a ZooKeeper Client**.

12.29 Appendix

12.29.1 Modifying Cluster Service Configuration Parameters

- You can modify service configuration parameters on the cluster management page of the MRS management console.
  a. Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click a cluster name.
  b. Choose **Components** > *Name of the desired service* > **Service Configuration**.

     The **Basic Configuration** tab page is displayed by default. To modify more parameters, click the **All Configurations** tab. The navigation tree displays all configuration parameters of the service. The level-1 nodes in the navigation tree are service names or role names. The parameter category is displayed after the level-1 node is expanded.
  c. In the navigation tree, select the specified parameter category and change the parameter values on the right.

     If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The system searches for the parameter in real time and displays the result.
  d. Click **Save Configuration**. In the displayed dialog box, click **OK**.
  e. Wait until the message **Operation successful** is displayed. Click **Finish**.

     The configuration is modified.

     Check whether there is any service whose configuration has expired in the cluster. If yes, restart the corresponding service or role instance for the configuration to take effect. You can also select **Restart the affected services or instances** when saving the configuration. .
- For MRS 3.*x* or earlier: You can log in to MRS Manager to modify service configuration parameters.

a. Log in to MRS Manager.
b. Click **Services**.
c. Click the specified service name on the service management page.
d. Click **Service Configuration**.

   The **Basic Configuration** tab page is displayed by default. To modify more parameters, click the **All Configurations** tab. The navigation tree displays all configuration parameters of the service. The level-1 nodes in the navigation tree are service names or role names. The parameter category is displayed after the level-1 node is expanded.
e. In the navigation tree, select the specified parameter category and change the parameter values on the right.

   If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The system searches for the parameter in real time and displays the result.
f. Click **Save**. In the confirmation dialog box, click **OK**.
g. Wait until the message **Operation successful** is displayed. Click **Finish**.

   The configuration is modified.

   Check whether there is any service whose configuration has expired in the cluster. If yes, restart the corresponding service or role instance for the configuration to take effect. You can also select **Restart the affected services or instances** when saving the configuration.

- For MRS 3.*x* or later: You can log in to FusionInsight Manager to modify service configuration parameters.
  a. You have logged in to FusionInsight Manager.
  b. Choose **Cluster** > **Service**.
  c. Click the specified service name on the service management page.
  d. Click **Configuration**.

     The **Basic Configuration** tab page is displayed by default. To modify more parameters, click the **All Configurations** tab. The navigation tree displays all configuration parameters of the service. The level-1 nodes in the navigation tree are service names or role names. The parameter category is displayed after the level-1 node is expanded.
  e. In the navigation tree, select the specified parameter category and change the parameter values on the right.

     If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The system searches for the parameter in real time and displays the result.
  f. Click **Save**. In the confirmation dialog box, click **OK**.
  g. Wait until the message **Operation successful** is displayed. Click **Finish**.

     The configuration is modified.

     Check whether there is any service whose configuration has expired in the cluster. If yes, restart the corresponding service or role instance for the configuration to take effect.

12.29.2 Accessing Manager

12.29.2.1 Accessing MRS Manager (Versions Earlier Than MRS 3.x)

Scenario

Clusters of versions earlier than MRS 3.x use MRS Manager to monitor, configure, and manage clusters. You can open the MRS Manager page on the MRS console.

Accessing MRS manager

Step 1 Log in to the MRS management console.

Step 2 In the navigation pane, choose **Clusters** > **Active Clusters**. Click the target cluster name to access the cluster details page.

Step 3 Click **Access Manager**. The **Access MRS Manager** page is displayed.

- If you have bound an EIP when creating a cluster,
  a. Select the security group to which the security group rule to be added belongs. The security group is configured when the cluster is created.
  b. Add a security group rule. By default, your public IP address used for accessing port 9022 is filled in the rule. To enable multiple IP address segments to access MRS Manager, see **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

     NOTE

     - It is normal that the automatically generated public IP address is different from the local IP address and no action is required.
     - If port 9022 is a Knox port, you need to enable the permission of port 9022 to access Knox for accessing MRS Manager.

  c. Select the checkbox stating that **I confirm that xx.xx.xx.xx is a trusted public IP address and MRS Manager can be accessed using this IP address**.
- If you have not bound an EIP when creating a cluster,
  a. Select an available EIP from the drop-down list or click **Manage EIP** to create one.
  b. Select the security group to which the security group rule to be added belongs. The security group is configured when the cluster is created.
  c. Add a security group rule. By default, your public IP address used for accessing port 9022 is filled in the rule. To enable multiple IP address segments to access MRS Manager, see **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

     NOTE

     - It is normal that the automatically generated public IP address is different from the local IP address and no action is required.
     - If port 9022 is a Knox port, you need to enable the permission of port 9022 to access Knox for accessing MRS Manager.

d. Select the checkbox stating that **I confirm that xx.xx.xx.xx is a trusted public IP address and MRS Manager can be accessed using this IP address**.

Step 4 Click **OK**. The MRS Manager login page is displayed.

Step 5 Enter the default username **admin** and the password set during cluster creation, and click **Log In**. The MRS Manager page is displayed.

Step 6 On the MRS console, click **Clusters** and choose **Active Clusters**. Click the target cluster name to access the cluster details page.

NOTE

To assign MRS Manager access permissions to other users, follow instructions from **Step 6** to **Step 9** to add the users' public IP addresses to the trusted range.

Step 7 Click **Add Security Group Rule** on the right of **EIP**.

Step 8 On the **Add Security Group Rule** page, add the IP address segment for users to access the public network and select **I confirm that the authorized object is a trusted public IP address range. Do not use 0.0.0.0/0. Otherwise, security risks may arise**.

By default, the IP address used for accessing the public network is filled. You can change the IP address segment as required. To enable multiple IP address segments, repeat steps **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

Step 9 Click **OK**.

----End

Granting the Permission to Access MRS Manager to Other Users

Step 1 On the MRS console, click **Clusters** and choose **Active Clusters**. Click the target cluster name to access the cluster details page.

Step 2 Click **Add Security Group Rule** on the right of **EIP**.

Step 3 On the **Add Security Group Rule** page, add the IP address segment for users to access the public network and select.**I confirm that the authorized object is a trusted public IP address range. Do not use 0.0.0.0/0. Otherwise, security risks may arise.**

By default, the IP address used for accessing the public network is filled. You can change the IP address segment as required. To enable multiple IP address segments, repeat steps **Step 1** to **Step 4**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

Step 4 Click **OK**.

----End

12.29.2.2 Accessing FusionInsight Manager (MRS 3.*x* or Later)

Scenario

In MRS 3.*x* or later, FusionInsight Manager is used to monitor, configure, and manage clusters. After the cluster is installed, you can use the account to log in to FusionInsight Manager.

NOTE

If you cannot log in to the WebUI of the component, access FusionInsight Manager by referring to **Accessing FusionInsight Manager from an ECS**.

Accessing FusionInsight Manager Using EIP

Step 1 Log in to the MRS management console.

Step 2 In the navigation pane, choose **Clusters** > **Active Clusters**. Click the target cluster name to access the cluster details page.

Step 3 Click **Manager** next to **MRS Manager**. In the displayed dialog box, configure the EIP information.

1. If no EIP is bound during MRS cluster creation, select an available EIP from the drop-down list on the right of **IEP**. If you have bound an EIP when creating a cluster, go to **Step 3.2**.

   NOTE

   If no EIP is available, click **Manage EIP** to one. Then, select the d EIP from the drop-down list on the right of **EIP**.

2. Select the security group to which the security group rule to be added belongs. The security group is configured when the cluster is created.
3. Add a security group rule. By default, the filled-in rule is used to access the EIP. To enable multiple IP address segments to access Manager, see steps **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.
4. Select the information to be confirmed and click **OK**.

Step 4 Click **OK**. The Manager login page is displayed.

Step 5 Enter the default username **admin** and the password set during cluster creation, and click **Log In**. The Manager page is displayed.

Step 6 On the MRS management console, choose **Clusters** > **Active Clusters**. Click the target cluster name to access the cluster details page.

NOTE

To grant other users the permission to access Manager, perform **Step 6** to **Step 9** to add the users' public IP addresses to the trusted IP address range.

Step 7 Click **Add Security Group Rule** on the right of **EIP**.

Step 8 On the **Add Security Group Rule** page, add the IP address segment for users to access the public network and select **I confirm that *public network IP*|*port* is a**

trusted public IP address. I understand that using 0.0.0.0/0. poses security risks.

By default, the IP address used for accessing the public network is filled. You can change the IP address segment as required. To enable multiple IP address segments, repeat steps **Step 6** to **Step 9**. If you want to view, modify, or delete a security group rule, click **Manage Security Group Rule**.

Step 9 Click **OK**.

----End

Accessing FusionInsight Manager from an ECS

Step 1 On the MRS management console, click **Clusters**.

Step 2 On the **Active Clusters** page, click the name of the specified cluster.

Record the **AZ**, **VPC**, **MRS ManagerSecurity Group** of the cluster.

Step 3 On the homepage of the management console, choose **Service List** > **Elastic Cloud Server** to switch to the ECS management console and create an ECS.

- The **AZ**, **VPC**, and **Security Group** of the ECS must be the same as those of the cluster to be accessed.
- Select a Windows public image. For example, a standard image **Windows Server 2012 R2 Standard 64bit(40GB)**.
- For details about other configuration parameters, see **Elastic Cloud Server > User Guide > Getting Started > Creating and Logging In to a Windows ECS**.

NOTE

If the security group of the ECS is different from **Default Security Group** of the Master node, you can modify the configuration using either of the following methods:

- Change the security group of the ECS to the default security group of the Master node. For details, see **Elastic Cloud Server** > **User Guide** > **Security Group** > **Changing a Security Group**.
- Add two security group rules to the security groups of the Master and Core nodes to enable the ECS to access the cluster. Set **Protocol** to **TCP**, **Ports** of the two security group rules to **28443** and **20009**, respectively. For details, see **Virtual Private Cloud > User Guide > Security > Security Group > Adding a Security Group Rule**.

Step 4 On the VPC management console, apply for an EIP and bind it to the ECS.

For details, see **Virtual Private Cloud** > **User Guide** > **Elastic IP** > **Assigning an EIP and Binding It to an ECS**.

Step 5 Log in to the ECS.

The Windows system account, password, EIP, and the security group rules are required for logging in to the ECS. For details, see **Elastic Cloud Server > User Guide > Instances > Logging In to a Windows ECS**.

Step 6 On the Windows remote desktop, use your browser to access Manager.

For example, you can use Internet Explorer 11 in the Windows 2012 OS.

The address for accessing Manager is the address of the **MRS Manager** page. Enter the name and password of the cluster user, for example, user **admin**.

NOTE

- If you access Manager with other cluster usernames, change the password upon your first access. The new password must meet the requirements of the current password complexity policies. For details, contact the administrator.
- By default, a user is locked after inputting an incorrect password five consecutive times. The user is automatically unlocked after 5 minutes.

----End

12.29.3 Using an MRS Client

12.29.3.1 Installing a Client (Version 3.x or Later)

Scenario

This section describes how to install clients of all services (excluding Flume) in an MRS cluster. For details about how to install the Flume client, see **Installing the Flume Client**.

A client can be installed on a node inside or outside the cluster. This section uses the installation directory **//opt/client** as an example. Replace it with the actual one.

Prerequisites

- A Linux ECS has been prepared. For details about the supported OS of the ECS, see **Table 12-480**.

  Table 12-480 Reference list

CPU Architecture	OS	Supported Version
x86 computing	Euler	Euler OS 2.5
	SuSE	SUSE Linux Enterprise Server 12 SP4 (SUSE 12.4)
	Red Hat	Red Hat-7.5-x86_64 (Red Hat 7.5)
	CentOS	CentOS 7.6
Kunpeng computing (Arm)	Euler	Euler OS 2.8
	CentOS	CentOS 7.6

  In addition, sufficient disk space is allocated for the ECS, for example, 40 GB.
- The ECS and the MRS cluster are in the same VPC.
- The security group of the ECS must be the same as that of the master node in the MRS cluster.

- The NTP service has been installed on the ECS OS and is running properly.

  If the NTP service is not installed, run the **yum install ntp -y** command to install it when the **yum** source is configured.
- A user can log in to the Linux ECS using the password (in SSH mode).

Installing a Client on a Node Inside a Cluster

1. Obtain the software package.

   Log in to FusionInsight Manager. For details, see **Accessing FusionInsight Manager (MRS 3.x or Later)**. Click the name of the cluster to be operated in the **Cluster** drop-down list.

   Choose **More > Download Client**. The **Download Cluster Client** dialog box is displayed.

   NOTE

   In the scenario where only one client is to be installed, choose **Cluster > Service >** *Service name* **> More > Download Client**. The **Download Client** dialog box is displayed.

2. Set the client type to **Complete Client**.

   Configuration Files Only is to download client configuration files in the following scenario: After a complete client is downloaded and installed and administrators modify server configurations on Manager, developers need to update the configuration files during application development.

   The platform type can be set to **x86_64** or **aarch64**.

   - **x86_64**: indicates the client software package that can be deployed on the x86 servers.
   - **aarch64**: indicates the client software package that can be deployed on the TaiShan servers.

   NOTE

   The cluster supports two types of clients: **x86_64** and **aarch64**. The client type must match the architecture of the node for installing the client. Otherwise, client installation will fail.

3. Select **Save to Path** and click **OK** to generate the client file.

   The generated file is stored in the **/tmp/FusionInsight-Client** directory on the active management node by default. You can also store the client file in a directory on which user **omm** has the read, write, and execute permissions. Copy the software package to the file directory on the server where the client is to be installed as user **omm** or **root**.

   The name of the client software package is in the follow format: **FusionInsight_Cluster_***<Cluster ID>***_Services_Client.tar**.

   The following steps and sections use **FusionInsight_Cluster_1_Services_Client.tar** as an example.

NOTE

If you cannot obtain the permissions of user **root**, use user **omm**.

To install the client on another node in the cluster, run the following command to copy the client to the node where the client is to be installed:

scp -p /*tmp/FusionInsight-Client***/FusionInsight_Cluster_1_Services_Client.tar** *IP address of the node where the client is to be installed:/opt/Bigdata/client*

4. Log in to the server where the client software package is located as user **user_client**.
5. Decompress the software package.

   Go to the directory where the installation package is stored, such as **/tmp/FusionInsight-Client**. Run the following command to decompress the installation package to a local directory:

   tar -xvf FusionInsight_Cluster_1_Services_Client.tar
6. Verify the software package.

   Run the following command to verify the decompressed file and check whether the command output is consistent with the information in the **sha256** file.

   sha256sum -c FusionInsight_Cluster_1_Services_ClientConfig.tar.sha256

   ```
   FusionInsight_Cluster_1_Services_ClientConfig.tar: OK
   ```
7. Decompress the obtained installation file.

   tar -xvf FusionInsight_Cluster_1_Services_ClientConfig.tar
8. Go to the directory where the installation package is stored, and run the following command to install the client to a specified directory (an absolute path), for example, **/opt/client**:

   cd /tmp/FusionInsight-ClientFusionInsight_Cluster_1_Services_ClientConfig

   Run the **./install.sh /opt/client** command to install the client. The client is successfully installed if information similar to the following is displayed:

   ```
   The component client is installed successfully
   ```

   NOTE

   - If the clients of all or some services use the **/opt/client** directory, other directories must be used when you install other service clients.
   - You must delete the client installation directory when uninstalling a client.
   - To ensure that an installed client can only be used by the installation user (for example, **user_client**), add parameter **-o** during the installation. That is, run the **./install.sh /opt/client -o** command to install the client.
   - If an HBase client is installed, it is recommended that the client installation directory contain only uppercase and lowercase letters, digits, and characters (_-?.@+=) due to the limitation of the Ruby syntax used by HBase.

Using a Client

1. On the node where the client is installed, run the **sudo su - omm** command to switch the user. Run the following command to go to the client directory:

   cd /opt/client
2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *MRS cluster user*

   Example: **kinit admin**

   NOTE

   User **admin** is created by default for MRS clusters with Kerberos authentication enabled and is used for administrators to maintain the clusters.

4. Run the client command of a component directly.

   For example, run the **hdfs dfs -ls /** command to view files in the HDFS root directory.

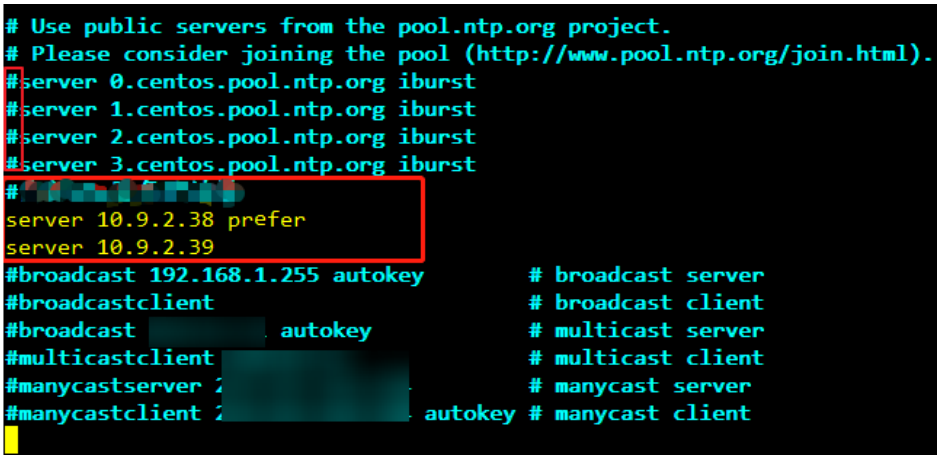
Installing a Client on a Node Outside a Cluster

1. Create an ECS that meets the requirements in **Prerequisites**.
2. Perform NTP time synchronization to synchronize the time of nodes outside the cluster with that of the MRS cluster.

   a. Run the **vi /etc/ntp.conf** command to edit the NTP client configuration file, add the IP addresses of the master node in the MRS cluster, and comment out the IP address of other servers.

   ```
   server master1_ip prefer
   server master2_ip
   ```

   Figure 12-75 Adding the master node IP addresses

   b. Run the **service ntpd stop** command to stop the NTP service.

   c. Run the **/usr/sbin/ntpdate** *IP address of the active master node* command to manually synchronize time.

   d. Run the **service ntpd start** or **systemctl restart ntpd** command to start the NTP service.

   e. Run the **ntpstat** command to check the time synchronization result.

3. Perform the following steps to download the cluster client software package from FusionInsight Manager, copy the package to the ECS node, and install the client:

   a. Log in to FusionInsight Manager and download the cluster client to the specified directory on the active management node by referring to

Accessing FusionInsight Manager (MRS 3.x or Later) and **Installing a Client on a Node Inside a Cluster**.

b. Log in to the active management node as user **root** and run the following command to copy the client installation package to the target node:

scp -p /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_Client.tar *IP address of the node where the client is to be installed*:**/tmp**

c. Log in to the node on which the client is to be installed as the client user.

Run the following commands to install the client. If the user does not have operation permissions on the client software package and client installation directory, grant the permissions using the **root** user.

cd /tmp

tar -xvf FusionInsight_Cluster_1_Services_Client.tar

tar -xvf FusionInsight_Cluster_1_Services_ClientConfig.tar

cd FusionInsight_Cluster_1_Services_ClientConfig

./install.sh /opt/client

d. Run the following commands to switch to the client directory and configure environment variables:

cd /opt/client

source bigdata_env

e. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

f. Run the client command of a component directly.

For example, run the **hdfs dfs -ls /** command to view files in the HDFS root directory.

12.29.3.2 Installing a Client (Versions Earlier Than 3.x)

Scenario

An MRS client is required. The MRS cluster client can be installed on the Master or Core node in the cluster or on a node outside the cluster.

After a cluster of versions earlier than MRS 3.x is created, a client is installed on the active Master node by default. You can directly use the client. The installation directory is **/opt/client**.

For details about how to install a client of MRS 3.x or later, see **Installing a Client (Version 3.x or Later)**.

NOTE

If a client has been installed on the node outside the MRS cluster and the client only needs to be updated, update the client using the user who installed the client, for example, user **root**.

Prerequisites

- An ECS has been prepared. For details about the OS and its version of the ECS, see **Table 12-481**.

  Table 12-481 Reference list

OS	Supported Version
EulerOS	• Available: EulerOS 2.2 • Available: EulerOS 2.3 • Available: EulerOS 2.5

  For example, a user can select an ECS running the EulerOS.

  In addition, sufficient disk space is allocated for the ECS, for example, 40 GB.
- The ECS and the MRS cluster are in the same VPC.
- The security group of the ECS is the same as that of the Master node of the MRS cluster.

  If this requirement is not met, modify the ECS security group or configure the inbound and outbound rules of the ECS security group to allow the ECS security group to be accessed by all security groups of MRS cluster nodes.
- To enable users to log in to a Linux ECS using a password (SSH), see ***Instances* > *Logging In to a Linux ECS* > *Login Using an SSH Password*** *in the Elastic Cloud Server User Guide*.

Installing a Client on the Core Node

1. Log in to MRS Manager and choose **Services** > **Download Client** to download the client installation package to the active management node.

   NOTE

   If only the client configuration file needs to be updated, see method 2 in **Updating a Client (Versions Earlier Than 3.x)**.
2. Use the IP address to search for the active management node, and log in to the active management node using VNC.
3. Log in to the active management node, and run the following command to switch the user:

   sudo su - omm
4. On the MRS management console, view the IP address on the **Nodes** tab page of the specified cluster.

   Record the IP address of the Core node where the client is to be used.
5. On the active management node, run the following command to copy the client installation package to the Core node:

   scp -p /tmp/MRS-client/MRS_Services_Client.tar *IP address of the Core node*:**/opt/client**
6. Log in to the Core node as user **root**.

Master nodes support Cloud-Init. The preset username for Cloud-Init is **root** and the password is the one you set during cluster creation.

7. Run the following commands to install the client:

   cd /opt/client

   tar -xvf MRS_Services_Client.tar

   tar -xvf MRS_Services_ClientConfig.tar

   cd /opt/client/MRS_Services_ClientConfig

   ./install.sh *Client installation directory*

   For example, run the following command:

   ./install.sh /opt/client

8. For details about how to use the client, see **Using an MRS Client**.

Using an MRS Client

1. On the node where the client is installed, run the **sudo su - omm** command to switch the user. Run the following command to go to the client directory:

   cd /opt/client

2. Run the following command to configure environment variables:

   source bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

   kinit *MRS cluster user*

   Example: **kinit admin**

   NOTE

   User **admin** is created by default for MRS clusters with Kerberos authentication enabled and is used for administrators to maintain the clusters.

4. Run the client command of a component directly.

   For example, run the **hdfs dfs -ls /** command to view files in the HDFS root directory.

Installing a Client on a Node Outside the Cluster

Step 1 Create an ECS that meets the requirements in the prerequisites.

Step 2 Log in to MRS Manager. For details, see **Accessing MRS Manager (Versions Earlier Than MRS 3.x)**. Then, choose **Services**.

Step 3 Click **Download Client**.

Step 4 In **Client Type**, select **All client files**.

Step 5 In **Download To**, select **Remote host**.

Step 6 Set **Host IP Address** to the IP address of the ECS, **Host Port** to **22**, and **Save Path** to **/tmp**.

- If the default port **22** for logging in to an ECS using SSH has been changed, set **Host Port** to the new port.

- **Save Path** contains a maximum of 256 characters.

Step 7 Set **Login User** to **root**.

If other users are used, ensure that the users have read, write, and execute permission on the save path.

Step 8 Select **Password** or **SSH Private Key** for **Login Mode**.

- **Password**: Enter the password of user **root** set during cluster creation.
- **SSH Private Key**: Select and upload the key file used for creating the cluster.

Step 9 Click **OK** to generate a client file.

If the following information is displayed, the client package is saved. Click **Close**. Obtain the client file from the save path on the remote host that is set when the client is downloaded.

```
Client files downloaded to the remote host successfully.
```

If the following information is displayed, check the username, password, and security group configurations of the remote host. Ensure that the username and password are correct and an inbound rule of the SSH (22) port has been added to the security group of the remote host. And then, go to **Step 2** to download the client again.

```
Failed to connect to the server. Please check the network connection or parameter settings.
```

NOTE

Generating a client will occupy a large number of disk I/Os. You are advised not to download a client when the cluster is being installed, started, and patched, or in other unstable states.

Step 10 Log in to the ECS using VNC. For details, see **Instance** > **Logging In to a Linux** > **Logging In to a Linux** in the *Elastic Cloud Server User Guide*

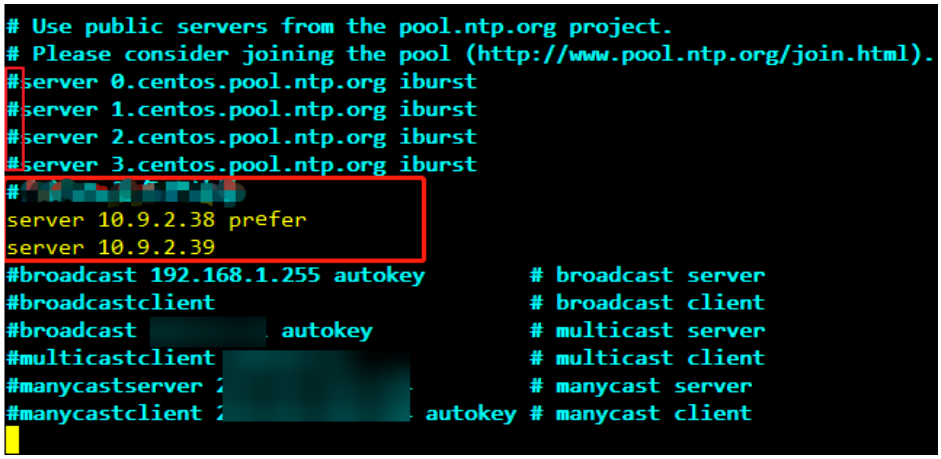
All images support Cloud-Init. The preset username for Cloud-Init is **root** and the password is the one you set during cluster creation. It is recommended that you change the password upon the first login.

Step 11 Perform NTP time synchronization to synchronize the time of nodes outside the cluster with the time of the MRS cluster.

1. Check whether the NTP service is installed. If it is not installed, run the **yum install ntp -y** command to install it.
2. Run the **vim /etc/ntp.conf** command to edit the NTP client configuration file, add the IP address of the Master node in the MRS cluster, and comment out the IP addresses of other servers.

   ```
   server master1_ip prefer
   server master2_ip
   ```

Figure 12-76 Adding the master node IP addresses



3. Run the **service ntpd stop** command to stop the NTP service.
4. Run the**/usr/sbin/ntpdate** *IP address of the active Master node* command to manually synchronize the time.
5. Run the **service ntpd start** or **systemctl restart ntpd** command to start the NTP service.
6. Run the **ntpstat** command to check the time synchronization result:

Step 12 On the ECS, switch to user **root** and copy the installation package in **Save Path** in **Step 6** to the **/opt** directory. For example, if **Save Path** is set to **/tmp**, run the following commands:

sudo su - root

cp /tmp/MRS_Services_Client.tar /opt

Step 13 Run the following command in the **/opt** directory to decompress the package and obtain the verification file and the configuration package of the client:

tar -xvf MRS_Services_Client.tar

Step 14 Run the following command to verify the configuration file package of the client:

sha256sum -c MRS_Services_ClientConfig.tar.sha256

The command output is as follows:

```
MRS_Services_ClientConfig.tar: OK
```

Step 15 Run the following command to decompress **MRS_Services_ClientConfig.tar**:

tar -xvf MRS_Services_ClientConfig.tar

Step 16 Run the following command to install the client to a new directory, for example, **/opt/Bigdata/client**. A directory is automatically generated during the client installation.

sh /opt/MRS_Services_ClientConfig/install.sh /opt/Bigdata/client

If the following information is displayed, the client has been successfully installed:

```
Components client installation is complete.
```

Step 17 Check whether the IP address of the ECS node is connected to the IP address of the cluster Master node.

For example, run the following command: **ping** *Master node IP address*.

- If yes, go to **Step 18**.
- If no, check whether the VPC and security group are correct and whether the ECS and the MRS cluster are in the same VPC and security group, and go to **Step 18**.

Step 18 Run the following command to configure environment variables:

source /opt/Bigdata/client/bigdata_env

Step 19 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 20 Run the client command of a component.

For example, run the following command to query the HDFS directory:

hdfs dfs -ls /

----End

12.29.3.3 Updating a Client (Version 3.x or Later)

A cluster provides a client for you to connect to a server, view task results, or manage data. If you modify service configuration parameters on Manager and restart the service, you need to download and install the client again or use the configuration file to update the client.

Updating the Client Configuration

Method 1:

Step 1 Log in to FusionInsight Manager. For details, see **Accessing MRS Manager (Versions Earlier Than MRS 3.x)**. Click the name of the cluster to be operated in the **Cluster** drop-down list.

Step 2 Choose **More** > **Download Client** > **Configuration Files Only**.

The generated compressed file contains the configuration files of all services.

Step 3 Determine whether to generate a configuration file on the cluster node.

- If yes, select **Save to Path**, and click **OK** to generate the client file. By default, the client file is generated in **/tmp/FusionInsight-Client** on the active management node. You can also store the client file in other directories, and user **omm** has the read, write, and execute permissions on the directories. Then go to **Step 4**.
- If no, click **OK**, specify a local save path, and download the complete client. Wait until the download is complete and go to **Step 4**.

Step 4 Use WinSCP to save the compressed file to the client installation directory, for example, **/opt/hadoopclient**, as the client installation user.

Step 5 Decompress the software package.

Run the following commands to go to the directory where the client is installed, and decompress the file to a local directory. For example, the downloaded client file is **FusionInsight_Cluster_1_Services_Client.tar**.

cd /opt/hadoopclient

tar -xvf FusionInsight_Cluster_1_Services_Client.tar

Step 6 Verify the software package.

Run the following command to verify the decompressed file and check whether the command output is consistent with the information in the **sha256** file.

sha256sum -c FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar.sha256

```
FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar: OK
```

Step 7 Decompress the package to obtain the configuration file.

tar -xvf FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles.tar

Step 8 Run the following command in the client installation directory to update the client using the configuration file:

sh refreshConfig.sh *Client installation directory Directory where the configuration file is located*

For example, run the following command:

sh refreshConfig.sh /opt/hadoopclient /opt/hadoopclient/FusionInsight_Cluster_1_Services_ClientConfig_ConfigFiles

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

----End

Method 2:

Step 1 Log in to the client installation node as user **root**.

Step 2 Go to the client installation directory, for example, **/opt/hadoopclient** and run the following commands to update the configuration file:

cd /opt/hadoopclient

sh autoRefreshConfig.sh

Step 3 Enter the username and password of the FusionInsight Manager administrator and the floating IP address of FusionInsight Manager.

Step 4 Enter the names of the components whose configuration needs to be updated. Use commas (,) to separate the component names. Press **Enter** to update the configurations of all components if necessary.

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

----End

12.29.3.4 Updating a Client (Versions Earlier Than 3.x)

NOTE

This section applies to clusters of versions earlier than MRS 3.x. For MRS 3.x or later, see **Updating a Client (Version 3.x or Later)**.

Updating a Client Configuration File

Scenario

An MRS cluster provides a client for you to connect to a server, view task results, or manage data. Before using an MRS client, you need to download and update the client configuration file if service configuration parameters are modified and a service is restarted or the service is merely restarted on MRS Manager.

During cluster creation, the original client is stored in the **/opt/client** directory on all nodes in the cluster by default. After the cluster is created, only the client of a Master node can be directly used. To use the client of a Core node, you need to update the client configuration file first.

Procedure

Method 1:

Step 1 Log in to MRS Manager. For details, see **Accessing MRS Manager (Versions Earlier Than MRS 3.x)**. Then, choose **Services**.

Step 2 Click **Download Client**.

Set **Client Type** to **Only configuration files**, **Download To** to **Server**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default. You can customize the file path.

Step 3 Query and log in to the active Master node.

Step 4 If you use the client in the cluster, run the following command to switch to user **omm**. If you use the client outside the cluster, switch to user **root**.

sudo su - omm

Step 5 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 6 Run the following command to update client configurations:

sh refreshConfig.sh *Client installation directory Full path of the client configuration file package*

For example, run the following command:

sh refreshConfig.sh /opt/Bigdata/client /tmp/MRS-client/MRS_Services_Client.tar

If the following information is displayed, the configurations have been updated successfully.

```
ReFresh components client config is complete.
Succeed to refresh components client config.
```

----End

Method 2:

Step 1 After the cluster is installed, run the following command to switch to user **omm**. If you use the client outside the cluster, switch to user **root**.

sudo su - omm

Step 2 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 3 Run the following command and enter the name of an MRS Manager user with the download permission and its password (for example, the username is **admin** and the password is the one set during cluster creation) as prompted to update client configurations.

sh autoRefreshConfig.sh

Step 4 After the command is executed, the following information is displayed, where *XXX* indicates the name of the component installed in the cluster. To update client configurations of all components, press **Enter**. To update client configurations of some components, enter the component names and separate them with commas (,).

```
Components "xxx" have been installed in the cluster. Please input the comma-separated names of the components for which you want to update client configurations. If you press Enter without inputting any component name, the client configurations of all components will be updated:
```

If the following information is displayed, the configurations have been updated successfully.

```
Succeed to refresh components client config.
```

If the following information is displayed, the username or password is incorrect.

```
login manager failed,Incorrect username or password.
```

NOTE

- This script automatically connects to the cluster and invokes the **refreshConfig.sh** script to download and update the client configuration file.
- By default, the client uses the floating IP address specified by **wsom=xxx** in the **Version** file in the installation directory to update the client configurations. To update the configuration file of another cluster, modify the value of **wsom=xxx** in the **Version** file to the floating IP address of the corresponding cluster before performing this step.

----End

Fully Updating the Original Client of the Active Master Node

Scenario

During cluster creation, the original client is stored in the **/opt/client** directory on all nodes in the cluster by default. The following uses **/opt/Bigdata/client** as an example.

- For a normal MRS cluster, you will use the pre-installed client on a Master node to submit a job on the management console page.
- You can also use the pre-installed client on the Master node to connect to a server, view task results, and manage data.

After installing the patch on the cluster, you need to update the client on the Master node to ensure that the functions of the built-in client are available.

Procedure

Step 1 Log in to MRS Manager. For details, see **Accessing MRS Manager (Versions Earlier Than MRS 3.x)**. Then, choose **Services**.

Step 2 Click **Download Client**.

Set **Client Type** to **All client files**, **Download To** to **Server**, and click **OK** to generate the client configuration file. The generated file is saved in the **/tmp/MRS-client** directory on the active management node by default. You can customize the file path.

Step 3 Query and log in to the active Master node.

Step 4 On the ECS, switch to user **root** and copy the installation package to the **/opt** directory.

sudo su - root

cp /tmp/MRS-client/MRS_Services_Client.tar /opt

Step 5 Run the following command in the **/opt** directory to decompress the package and obtain the verification file and the configuration package of the client:

tar -xvf MRS_Services_Client.tar

Step 6 Run the following command to verify the configuration file package of the client:

sha256sum -c MRS_Services_ClientConfig.tar.sha256

The command output is as follows:

```
MRS_Services_ClientConfig.tar: OK
```

Step 7 Run the following command to decompress **MRS_Services_ClientConfig.tar**:

tar -xvf MRS_Services_ClientConfig.tar

Step 8 Run the following command to move the original client to the **/opt/Bigdata/client_bak** directory:

mv /opt/Bigdata/client /opt/Bigdata/client_bak

Step 9 Run the following command to install the client in a new directory. The client path must be **/opt/Bigdata/client**.

sh /opt/MRS_Services_ClientConfig/install.sh /opt/Bigdata/client

If the following information is displayed, the client has been successfully installed:

```
Components client installation is complete.
```

Step 10 Run the following command to modify the user and user group of the **/opt/Bigdata/client** directory:

chown omm:wheel /opt/Bigdata/client -R

Step 11 Run the following command to configure environment variables:

source /opt/Bigdata/client/bigdata_env

Step 12 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Example: **kinit admin**

Step 13 Run the client command of a component.

For example, run the following command to query the HDFS directory:

hdfs dfs -ls /

----End

Fully Updating the Original Client of the Standby Master Node

Step 1 Repeat **Step 1** to **Step 3** to log in to the standby Master node, and run the following command to switch to user **omm**:

sudo su - omm

Step 2 Run the following command on the standby master node to copy the downloaded client package from the active master node:

scp omm@*master1 nodeIP address***:/tmp/MRS-client/MRS_Services_Client.tar /tmp/MRS-client/**

NOTE

- In this command, **master1** node is the active master node.
- **/tmp/MRS-client/** is an example target directory of the standby master node.

Step 3 Repeat **Step 4** to **Step 13** to update the client of the standby Master node.

----End

13 Security Description

13.1 Security Configuration Suggestions for Clusters with Kerberos Authentication Disabled

The Hadoop community version provides two authentication modes: Kerberos authentication (security mode) and Simple authentication (normal mode). When creating a cluster, you can choose to enable or disable Kerberos authentication.

Clusters in security mode use the Kerberos protocol for security authentication.

In normal mode, MRS cluster components use a native open source authentication mechanism, which is typically Simple authentication. If Simple authentication is used, authentication is automatically performed by a client user (for example, user **root**) by default when a client connects to a server. The authentication is imperceptible to the administrator or service user. In addition, when being executed, the client may even pretend to be any user (including **superuser**) by injecting **UserGroupInformation**. Cluster resource management and data control APIs are not authenticated on the server and are easily exploited and attacked by hackers.

Therefore, in normal mode, network access permissions must be strictly controlled to ensure cluster security. You are advised to perform the following operations to ensure cluster security.

- Deploy service applications on ECSs in the same VPC and subnet and avoid accessing MRS clusters through an external network.
- Configure security group rules to strictly control the access scope. Do not configure access rules that allow **Any** or **0.0.0.0** for the inbound direction of MRS cluster ports.
- If you want to access the native pages of the components in the cluster from the external, follow instructions in **Creating an SSH Channel for Connecting to an MRS Cluster and Configuring the Browser** for configuration.

13.2 Security Authentication Principles and Mechanisms

Function

For clusters in security mode with Kerberos authentication enabled, security authentication is required during application development.

Kerberos, is now used to a concept in authentication. The Kerberos protocol adopts a client–server model and cryptographic algorithms such as AES (Advanced Encryption Standard). It provides mutual authentication, that is, both the client and the server can verify each other's identity. Kerberos is used to prevent interception and replay attacks and protect data integrity. It is a system that manages keys by using a symmetric key mechanism.

Architecture

Kerberos architecture is shown in **Figure 13-1** and module description in **Table 13-1**.

Figure 13-1 Kerberos architecture

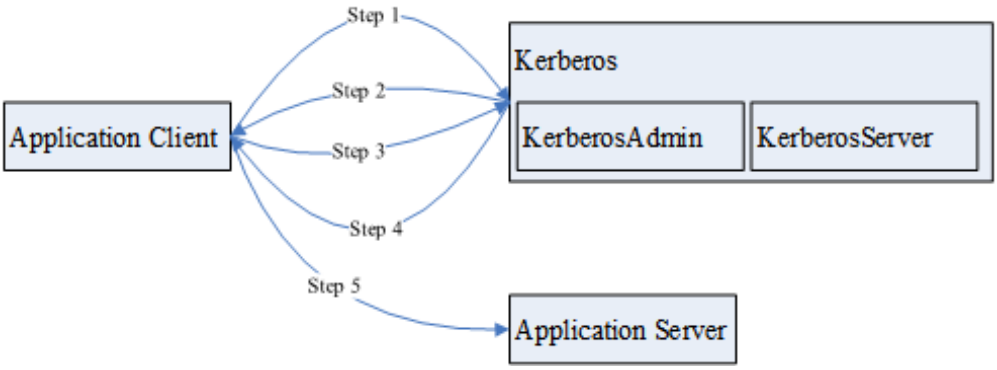


Table 13-1 Module description

Module	Description
Application Client	An application client, which is usually an application that submits tasks or jobs
Application Server	An application server, which is usually an application that an application client accesses
Kerberos	A service that provides security authentication

Module	Description
KerberosAdmin	A process that provides authentication user management
KerberosServer	A process that provides authentication ticket distribution

The process and principle are described as follows:

An application client can be a service in a cluster or a secondary development application of the customer. An application client can submit tasks or jobs to an application service.

1. Before submitting a task or job, the application client needs to apply for a ticket granting ticket (TGT) from the Kerberos service to establish a secure session with the Kerberos server.
2. After receiving the TGT request, the Kerberos service resolves parameters in the request to generate a TGT, and uses the key of the username specified by the client to encrypt the response.
3. After receiving the TGT response, the application client (based on the underlying RPC) resolves the response and obtains the TGT, and then applies for a server ticket (ST) of the application server from the Kerberos service.
4. After receiving the ST request, the Kerberos service verifies the TGT validity in the request and generates an ST of the application service, and then uses the application service key to encrypt the response.
5. After receiving the ST response, the application client packages the ST into a request and sends the request to the application server.
6. After receiving the request, the application server uses its local application service key to resolve the ST. After successful verification, the request becomes valid.

Basic Concepts

The following concepts can help users learn the Kerberos architecture quickly and understand the Kerberos service better. The following uses security authentication for HDFS as an example.

TGT

A TGT is generated by the Kerberos service and used to establish a secure session between an application and the Kerberos server. The validity period of a TGT is 24 hours. After 24 hours, the TGT expires automatically.

The following describes how to apply for a TGT (HDFS is used as an example):

1. Obtain a TGT through an API provided by HDFS.

```
/**
 * login Kerberos to get TGT, if the cluster is in security mode
 * @throws IOException if login is failed
 */
 private void login() throws IOException {
 // not security mode, just return
   if (! "kerberos".equalsIgnoreCase(conf.get("hadoop.security.authentication"))) {
```

```
        return;
    }

    //security mode
    System.setProperty("java.security.krb5.conf", PATH_TO_KRB5_CONF);

    UserGroupInformation.setConfiguration(conf);
    UserGroupInformation.loginUserFromKeytab(PRNCIPAL_NAME, PATH_TO_KEYTAB);
  }
```

2. Run shell commands on the client in kinit mode.

ST

An ST is generated by the Kerberos service and used to establish a secure session between an application and application service. An ST is valid only once.

In FusionInsight products, the generation of an ST is based on the Hadoop-RPC communication. The underlying RPC submits a request to the Kerberos server and the Kerberos server generates an ST.

Sample Authentication Code

```
package com.xxx.bigdata.hdfs.examples;

import java.io.IOException;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileStatus;
import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.security.UserGroupInformation;

public class KerberosTest {
    private static String PATH_TO_HDFS_SITE_XML = KerberosTest.class.getClassLoader().getResource("hdfs-
site.xml")
            .getPath();
    private static String PATH_TO_CORE_SITE_XML = KerberosTest.class.getClassLoader().getResource("core-
site.xml")
            .getPath();
    private static String PATH_TO_KEYTAB =
KerberosTest.class.getClassLoader().getResource("user.keytab").getPath();
    private static String PATH_TO_KRB5_CONF =
KerberosTest.class.getClassLoader().getResource("krb5.conf").getPath();
    private static String PRNCIPAL_NAME = "develop";
    private FileSystem fs;
    private Configuration conf;

    /**
     * initialize Configuration
     */
    private void initConf() {
        conf = new Configuration();

        // add configuration files
        conf.addResource(new Path(PATH_TO_HDFS_SITE_XML));
        conf.addResource(new Path(PATH_TO_CORE_SITE_XML));
    }

    /**
     * login Kerberos to get TGT, if the cluster is in security mode
     * @throws IOException if login is failed
     */
    private void login() throws IOException {
        // not security mode, just return
        if (! "kerberos".equalsIgnoreCase(conf.get("hadoop.security.authentication"))) {
            return;
        }
```

```
        //security mode
        System.setProperty("java.security.krb5.conf", PATH_TO_KRB5_CONF);

        UserGroupInformation.setConfiguration(conf);
        UserGroupInformation.loginUserFromKeytab(PRNCIPAL_NAME, PATH_TO_KEYTAB);
    }

    /**
     * initialize FileSystem, and get ST from Kerberos
     * @throws IOException
     */
    private void initFileSystem() throws IOException {
        fs = FileSystem.get(conf);
    }

    /**
     * An example to access the HDFS
     * @throws IOException
     */
    private void doSth() throws IOException {
        Path path = new Path("/tmp");
        FileStatus fStatus = fs.getFileStatus(path);
        System.out.println("Status of " + path + " is " + fStatus);
        //other thing
    }


    public static void main(String[] args) throws Exception {
        KerberosTest test = new KerberosTest();
        test.initConf();
        test.login();
        test.initFileSystem();
        test.doSth();
    }
}
```

NOTE

1. During Kerberos authentication, you need to configure the file parameters required for configuring the Kerberos authentication, including the keytab path, Kerberos authentication username, and the **krb5.conf** configuration file of the client for Kerberos authentication.
2. Method **login()** indicates calling the Hadoop API to perform Kerberos authentication and generating a TGT.
3. Method **doSth** indicates calling the Hadoop API to access the file system. In this situation, the underlying RPC automatically carries the TGT to Kerberos for verification and then an ST is generated.

14 High-Risk Operations Overview

Forbidden Operations

Table 14-1 lists forbidden operations during the routine cluster operation and maintenance process.

Table 14-1 Forbidden operations

Item	Risk
Delete ZooKeeper data directories.	ClickHouse, HDFS, Yarn, HBase, and Hive depend on ZooKeeper, which stores metadata. This operation has adverse impact on normal operating of related components.
Performing switchover frequently between active and standby JDBCServer nodes	This operation may interrupt services.
Delete Phoenix system tables and data (SYSTEM.CATALOG, SYSTEM.STATS, SYSTEM.SEQUENCE, and SYSTEM.FUNCTION).	This operation will cause service operation failures.
Manually modify data in the Hive metabase (hivemeta database).	This operation may cause Hive data parse errors. As a result, Hive cannot provide services.
Change permission on the Hive private file directory **hdfs:///tmp/hive-scratch**.	This operation may cause unavailable Hive services.
Modify **broker.id** in the Kafka configuration file.	This operation may cause invalid node data.
Modify the host names of nodes.	Instances and upper-layer components on the host cannot provide services properly. The fault cannot be rectified.

Item	Risk
Reinstall the OS of a node.	This operation will cause MRS cluster exceptions, leaving MRS clusters in abnormal status.
Use private images.	This operation will cause MRS cluster exceptions, leaving MRS clusters in abnormal status.

The following tables list the high-risk operations during the operation and maintenance of each component.

High-Risk Operations on a Cluster

Table 14-2 High-risk operations on a cluster

Operation	Risk	Severity	Workaround	Check Item
Modify the file directory or file permissions of user **omm** without permission.	This operation will lead to MRS service unavailability.	▲ ▲ ▲ ▲ ▲	Do not perform this operation.	Check whether the MRS cluster service is available.
Bind an EIP.	This operation exposes the Master node hosting MRS Manager of the cluster to the public network, increasing the risk of network attacks from the Internet.	▲ ▲ ▲ ▲ ▲	Ensure that the bound EIP is a trusted public IP address.	None

Operation	Risk	Severity	Workaround	Check Item
Enable security group rules for port 22 of a cluster.	This operation increases the risk of exploiting vulnerability of port 22.	▲ ▲ ▲ ▲ ▲	Configure a security group rule for port 22 to allow only trusted IP addresses to access the port. You are not advised to configure the inbound rule to allow 0.0.0.0 to access the port.	None
Delete a cluster or cluster data.	Data will get lost.	▲ ▲ ▲ ▲ ▲	Before deleting the data, confirm the necessity of the operation and ensure that the data has been backed up.	None
Scale in a cluster.	Data will get lost.	▲ ▲ ▲ ▲ ▲	Before scaling in the cluster, confirm the necessity of the operation and ensure that the data has been backed up.	None
Detach or format a data disk.	Data will get lost.	▲ ▲ ▲ ▲ ▲	Before performing this operation, confirm the necessity of the operation and ensure that the data has been backed up.	None

Manager High-Risk Operations

Table 14-3 Manager high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Change the OMS password.	This operation will restart all processes of OMSServer, which has adverse impact on cluster maintenance and management.	▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check whether there are uncleared alarms and whether the cluster management and maintenance are normal.
Import the certificate.	This operation will restart OMS processes and the entire cluster, which has adverse impact on cluster maintenance and management and services.	▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Perform an upgrade.	This operation will restart Manager and the entire cluster, affecting management, maintenance, and services of the cluster. Strictly manage the user who is eligible to assign the cluster management permission to prevent security risks.	▲ ▲ ▲	Ensure that there is no other maintenance and management operations when the operation is performed.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.

Operation	Risk	Severity	Workaround	Check Item
Restore the OMS.	This operation will restart Manager and the entire cluster, affecting management, maintenance, and services of the cluster.	▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Change an IP address.	This operation will restart Manager and the entire cluster, affecting management, maintenance, and services of the cluster.	▲ ▲ ▲	Ensure that there is no other maintenance and management operations when the operation is performed and that the new IP address is correct.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Change log levels.	If the log level is changed to **DEBUG**, Manager responds slowly.	▲ ▲	Before the modification, confirm the necessity of the operation and change it back to the default log level in time.	None

Operation	Risk	Severity	Workaround	Check Item
Replace a control node.	This operation will interrupt services deployed on the node. If the node is a management node, the operation will restart all OMS processes, affecting the cluster management and maintenance.	▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Replace a management node.	This operation will interrupt services deployed on the node. As a result, OMS processes will be restarted, affecting the cluster management and maintenance.	▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Restart the upper-layer service at the same time during the restart of a lower-layer service.	This operation will interrupt the upper-layer service, affecting the management, maintenance, and services of the cluster.	▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.

Operation	Risk	Severity	Workaround	Check Item
Modify the OLDAP port.	This operation will restart the LdapServer and Kerberos services and all associated services, affecting service running.	▲ ▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	None
Delete the **supergroup** group.	Deleting the **supergroup** group decreases user rights, affecting service access.	▲ ▲ ▲ ▲ ▲	Before the change, confirm the rights to be added. Ensure that the required rights have been added before deleting the **supergroup** rights to which the user is bound, ensuring service continuity.	None
Restart a service.	Services will be interrupted during the restart. If you select and restart the upper-layer service, the upper-layer services that depend on the service will be interrupted.	▲ ▲ ▲	Confirm the necessity of restarting the system before the operation.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.

Operation	Risk	Severity	Workaround	Check Item
Change the default SSH port No.	After the default port (22) is changed, functions such as cluster creation, service/instance adding, host adding, and host reinstallation cannot be used, and results of cluster health check items for node mutual trust, **omm**/**ommdba** user password expiration, and others are incorrect.	▲ ▲ ▲	Before performing this operation, restore the SSH port to the default value.	None

ClickHouse High-Risk Operations

Table 14-4 ClickHouse high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Delete data directories.	This operation may cause service information loss.	▲ ▲ ▲	Do not delete data directories manually.	Check whether data directories are normal.
Remove ClickHouseServer instances.	The ClickHouseServer instance nodes in the same shard must be removed in at the same time. Otherwise, the topology information of the logical cluster is incorrect. Before performing this operation, check the database and data table information of each node in the logical cluster and perform scale-in pre-analysis to ensure that data is successfully migrated during the scale-in process to prevent data loss	▲ ▲ ▲ ▲ ▲	Before scale-in, collect information in advance to learn the status of the ClickHouse logical cluster and instance nodes.	Check the ClickHouse logical cluster topology information, database and data table information in each ClickHouseServer instance, and data volume.

Operation	Risk	Severity	Workaround	Check Item
Add ClickHouseServer instances.	When performing this operation, you must check whether a database or data table with the same name as that on the old node needs to be created on the new node. Otherwise, subsequent data migration, data balancing, scale-in, and decommissioning will fail.	▲ ▲ ▲ ▲ ▲	Before scale-out, confirm the function and purpose of new ClickHouseServer instances and determine whether to create related databases and data tables.	Check the ClickHouse logical cluster topology information, database and data table information in each ClickHouseServer instance, and data volume.
Decommission ClickHouseServer instances.	The ClickHouseServer instance nodes in the same shard must be decommissioned in at the same time. Otherwise, the topology information of the logical cluster is incorrect. Before performing this operation, check the database and data table information of each node in the logical cluster and perform decommissioning pre-analysis to ensure that data is successfully migrated during the decommissioning process to prevent data loss	▲ ▲ ▲ ▲ ▲	Before decommissioning, collect information in advance to learn the status of the ClickHouse logical cluster and instance nodes.	Check the ClickHouse logical cluster topology information, database and data table information in each ClickHouseServer instance, and data volume.
Recommission ClickHouseServer instances.	When performing this operation, you must select all nodes in the original shard. Otherwise, the topology information of the logical cluster is incorrect.	▲ ▲ ▲ ▲ ▲	Before recommissioning, you need to confirm the home information about the shards of the node to be recommissioned.	Check the ClickHouse logical cluster topology information.

Operation	Risk	Severity	Workaround	Check Item
Modify data directory content (file and folder creation).	This operation may cause the ClickHouse instance of the node faults.	▲ ▲ ▲	Do not create or modify files or folders in the data directories manually.	Check whether data directories are normal.
Start or stop basic components independently.	This operation has adverse impact on the basic functions of some services. As a result, service failures occur.	▲ ▲ ▲	Do not start or stop ZooKeeper, Kerberos, and LDAP basic components independently. Select related services when performing this operation.	Check whether the service status is normal.
Restart or stop services.	This operation may interrupt services.	▲ ▲	Restart or stop services when necessary.	Check whether the service is running properly.

DBService High-Risk Operations

Table 14-5 DBService high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Change the DBService password.	The services need to be restarted for the password change to take effect. The services are unavailable during the restart.	▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check whether there are uncleared alarms and whether the cluster management and maintenance are normal.
Restore DBService data.	After the data is restored, the data generated after the data backup and before the data restoration is lost. After the data is restored, the configuration of the components that depend on DBService may expire and these components need to be restarted.	▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check whether there are uncleared alarms and whether the cluster management and maintenance are normal.

Operation	Risk	Severity	Workaround	Check Item
Perform active/standby DBService switchover.	During the DBServer switchover, DBService is unavailable.	▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	None
Change the DBService floating IP address.	The DBService needs to be restarted for the change to take effect. The DBService is unavailable during the restart. If the floating IP address has been used, the configuration will fail, and the DBService will fail to be started.	▲ ▲ ▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.

Flink High-Risk Operations

Table 14-6 Flink high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Change log levels.	If the log level is modified to DEBUG, the task running performance is affected.	▲ ▲	Before the modification, confirm the necessity of the operation and change it back to the default log level in time.	None

Operation	Risk	Severity	Workaround	Check Item
Modify file permissions.	Tasks may fail.	▲ ▲ ▲	Confirm the necessity of the operation before the modification.	Check whether related service operations are normal.

Flume High-Risk Operations

Table 14-7 Flume high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify the Flume instance start parameter **GC_OPTS**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Change the default value of **dfs.replication** from **3** to **1**.	This operation will have the following impacts: 1. The storage reliability deteriorates. If the disk becomes faulty, data will be lost. 2. NameNode fails to be restarted, and the HDFS service is unavailable.	▲ ▲ ▲ ▲	When modifying related configuration items, check the parameter description carefully. Ensure that there are more than two replicas for data storage.	Check whether the default replica number is not 1 and whether the HDFS service is normal.

HBase High-Risk Operations

Table 14-8 HBase high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify encryption configuration. • hbase.regionserver.wal.encryption • hbase.crypto.keyprovider.parameters.uri • hbase.crypto.keyprovider.parameters.encryptedtext	Services cannot start properly.	▲ ▲ ▲ ▲	Strictly follow the prompt information when modifying related configuration items, which are associated. Ensure that new values are valid.	Check whether services can be started properly.

Operation	Risk	Severity	Workaround	Check Item
Change the value of **hbase.regionserver.wal.encryption** to **false** or switch encryption algorithm from AES to SMS4.	This operation may cause start failures and data loss.	▲ ▲ ▲ ▲	When HFile and WAL are encrypted using an encryption algorithm and a table is created, do not close or switch the encryption algorithm randomly. If an encryption table (ENCRYPTION=>AES/SMS4) is not created, you can only switch the encryption algorithm.	None
Modify HBase instance start parameter **GC_OPTS** and **HBASE_HEAPSIZE**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid. GC_OPTS does not conflict with HBASE_HEAPSIZE.	Check whether services can be started properly.
Use **OfflineMetaRepair** tool	Services cannot start properly.	▲ ▲ ▲ ▲	This tool can be used only when HBase is offline and cannot be used in data migration scenarios.	Check whether HBase services can be started properly.

HDFS High-Risk Operations

Table 14-9 HDFS high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Change HDFS NameNode data storage directory **dfs.namenode.name.dir** and data configuration directory **dfs.datanode.data.dir**.	Services cannot start properly.	▲ ▲ ▲ ▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Use the **-delete** parameter when you run the **hadoop distcp** command.	During DistCP copying, files that do not exist in the source cluster but exist in the destination cluster are deleted from the destination cluster.	▲ ▲	When using DistCP, determine whether to retain the redundant files in the destination cluster. Exercise caution when using the **-delete** parameter.	After DistCP copying is complete, check whether the data in the destination cluster is retained or deleted according to the parameter settings.

Operation	Risk	Severity	Workaround	Check Item
Modify the HDFS instance start parameter **GC_OPTS**, **HADOOP_HEAPSIZE**, and **GC_PROFILE**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid. GC_OPTS does not conflict with HADOOP_HEAPSIZE.	Check whether services can be started properly.
Change the default value of **dfs.replication** from **3** to **1**.	This operation will have the following impacts: 1. The storage reliability deteriorates. If the disk becomes faulty, data will be lost. 2. NameNode fails to be restarted, and the HDFS service is unavailable.	▲ ▲ ▲ ▲	When modifying related configuration items, check the parameter description carefully. Ensure that there are more than two replicas for data storage.	Check whether the default replica number is not 1 and whether the HDFS service is normal.
Change the remote procedure call (RPC) channel encryption mode (**hadoop.rpc.protection**) of each module in Hadoop.	This operation causes service faults and service exceptions.	▲ ▲ ▲ ▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether HDFS and other services that depend on HDFS can properly start and provide services.

Hive High-Risk Operations

Table 14-10 Hive high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify the Hive instance start parameter **GC_OPTS**.	This operation may cause Hive instance start failures.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Delete all MetaStore instances.	This operation may cause Hive metadata loss. As a result, Hive cannot provide services.	▲ ▲ ▲	Do not perform this operation unless ensure that Hive table information can be discarded.	Check whether services can be started properly.
Delete or modify files corresponding to Hive tables over HDFS interfaces or HBase interfaces.	This operation may cause Hive service data loss or tampering.	▲ ▲	Do not perform this operation unless ensure that the data can be discarded or that the operation meets service requirements.	Check whether Hive data is complete.

Operation	Risk	Severity	Workaround	Check Item
Delete or modify files corresponding to Hive tables or directory access permission over HDFS interfaces or HBase interfaces.	This operation may cause related service scenarios to be unavailable.	▲ ▲ ▲	Do not perform this operation.	Check whether related service operations are normal.
Delete or modify **hdfs:///apps/templeton/hive-3.1.0.tar.gz** over HDFS interfaces.	WebHCat fails to perform services due to this operation.	▲ ▲	Do not perform this operation.	Check whether related service operations are normal.
Export table data to overwrite the data at the local. For example, export the data of **t1** to **/opt/dir**. **insert overwrite local directory '/opt/dir' select * from t1;**	This operation will delete target directories. Incorrect setting may cause software or OS startup failures.	▲ ▲ ▲ ▲ ▲	Ensure that the path where the data is written does not contain any files or do not use the key word overwrite in the command.	Check whether files in the target path are lost.

Operation	Risk	Severity	Workaround	Check Item
Direct different databases, tables, or partition files to the same path, for example, default warehouse path **/user/hive/warehouse**.	The creation operation may cause disordered data. After a database, table, or partition is deleted, other object data will be lost.	▲ ▲ ▲ ▲ ▲	Do not perform this operation.	Check whether files in the target path are lost.

Kafka High-Risk Operations

Table 14-11 Kafka high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Delete Topic	This operation may delete existing topics and data.	▲ ▲ ▲	Kerberos authentication is used to ensure that authenticated users have operation permissions. Ensure that topic names are correct.	Check whether topics are processed properly.
Delete data directories.	This operation may cause service information loss.	▲ ▲ ▲	Do not delete data directories manually.	Check whether data directories are normal.

Operation	Risk	Severity	Workaround	Check Item
Modify data directory content (file and folder creation).	This operation may cause the Broker instance of the node faults.	▲ ▲ ▲	Do not create or modify files or folders in the data directories manually.	Check whether data directories are normal.
Modify the disk auto-adaptation function using the **disk.adapter.enable** parameter.	This operation adjusts the topic data retention period when the disk usage reaches the threshold. Historical data that does not fall within the storage retention may be deleted.	▲ ▲ ▲	If the retention period of some topics cannot be adjusted, add this topic to the value of **disk.adapter.topic.blacklist**.	Observe the data storage period on the Kafka topic monitoring page.
Modify data directory **log.dirs** configuration.	Incorrect operation may cause process faults.	▲ ▲ ▲	Ensure that the added or modified data directories are empty and that the directory permissions are right.	Check whether data directories are normal.
Reduce the capacity of the Kafka cluster.	This operation may cause quantity reduction of backups of some data duplicates of topic. As a result, some topics cannot be accessed.	▲ ▲	Perform backup operation and then reduce the capacity of the Kafka cluster.	Check whether backup nodes where partitions are located are activated to ensure data security.

Operation	Risk	Severity	Workaround	Check Item
Start or stop basic components independently.	This operation has adverse impact on the basic functions of some services. As a result, service failures occur.	▲ ▲ ▲	Do not start or stop ZooKeeper, Kerberos, and LDAP basic components independently. Select related services when performing this operation.	Check whether the service status is normal.
Restart or stop services.	This operation may interrupt services.	▲ ▲	Restart or stop services when necessary.	Check whether the service is running properly.
Modify configuration parameters.	This operation requires service restart for configuration to take effect.	▲ ▲	Modify configuration when necessary.	Check whether the service is running properly.
Delete or modify metadata.	Modifying or deleting Kafka metadata on ZooKeeper may cause the Kafka topic or service unavailability.	▲ ▲ ▲	Do not delete or modify Kafka metadata stored on ZooKeeper.	Check whether the Kafka topics or Kafka service is available.
Delete metadata backup files.	After Kafka metadata backup files are modified and used to restore Kafka metadata, Kafka topics or the Kafka service may be unavailable.	▲ ▲ ▲	Do not delete Kafka metadata backup files.	Check whether the Kafka topics or Kafka service is available.

KrbServer High-Risk Operations

Table 14-12 KrbServer high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify the **KADMIN_PORT** parameter of KrbServer.	After this parameter is modified, if the KrbServer service and its associated services are not restarted in a timely manner, the configuration of KrbClient in the cluster is abnormal and the service running is affected.	▲ ▲ ▲ ▲ ▲	After this parameter is modified, restart the KrbServer service and all its associated services.	None
Modify the **kdc_ports** parameter of KrbServer.	After this parameter is modified, if the KrbServer service and its associated services are not restarted in a timely manner, the configuration of KrbClient in the cluster is abnormal and the service running is affected.	▲ ▲ ▲ ▲ ▲	After this parameter is modified, restart the KrbServer service and all its associated services.	None
Modify the **KPASSWD_PORT** parameter of KrbServer.	After this parameter is modified, if the KrbServer service and its associated services are not restarted in a timely manner, the configuration of KrbClient in the cluster is abnormal and the service running is affected.	▲ ▲ ▲ ▲ ▲	After this parameter is modified, restart the KrbServer service and all its associated services.	None

Operation	Risk	Severity	Workaround	Check Item
Modify the domain name of Manager system.	After the domain name is modified, if the KrbServer service and its associated services are not restarted in a timely manner, the configuration of KrbClient in the cluster is abnormal and the service running is affected.	▲ ▲ ▲ ▲ ▲	After this parameter is modified, restart the KrbServer service and all its associated services.	None
Configure cross-cluster mutual trust relationships.	This operation will restart the KrbServer service and all associated services, affecting the management and maintenance and services of the cluster.	▲ ▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.

LdapServer High-Risk Operations

Table 14-13 LdapServer high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify the **LDAP_SERVER_PORT** parameter of LdapServer.	After this parameter is modified, if the LdapServer service and its associated services are not restarted in a timely manner, the configuration of LdapClient in the cluster is abnormal and the service running is affected.	▲ ▲ ▲ ▲ ▲	After this parameter is modified, restart the LdapServer service and all its associated services.	None

Operation	Risk	Severity	Workaround	Check Item
Restore LdapServer data.	This operation will restart Manager and the entire cluster, affecting management, maintenance, and services of the cluster.	▲ ▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Replace the Node where LdapServer is located.	This operation will interrupt services deployed on the node. If the node is a management node, the operation will restart all OMS processes, affecting the cluster management and maintenance.	▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	Check for uncleared alarms, and check whether the cluster management and maintenance and services are normal.
Change the password of LdapServer.	The LdapServer and Kerberos services need to be restarted during the password change, affecting the management, maintenance, and services of the cluster.	▲ ▲ ▲ ▲	Before performing the operation, ensure that the operation is necessary, and that no other management and maintenance operations are performed at the same time.	None

Operation	Risk	Severity	Workaround	Check Item
Restart the node where LdapServer is located.	Restarting the node without stopping the LdapServer service may cause LdapServer data damage.	▲ ▲ ▲ ▲ ▲	Restore LdapServer using LdapServer backup data	None

Loader High-Risk Operations

Table 14-14 Loader high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Change the floating IP address of a Loader instance (**loader.float.ip**).	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether the Loader UI can be connected properly.
Modify the Loader instance start parameter **LOADER_GC_OPTS**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Clear table contents when adding data to HBase.	This operation will clear original data in the target table.	▲ ▲	Ensure that the contents in the target table can be cleared before the operation.	Check whether the contents in the target table can be cleared before the operation.

Spark2x High-risk Operations

NOTE

Spark high-risk operations apply to MRS 3.x earlier versions.

Table 14-15 Spark2x high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify the configuration item **spark.yarn.queue**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Modify the configuration item **spark.driver.extraJavaOptions**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Modify the configuration item **spark.yarn.cluster.driver.extraJavaOptions**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.

Operation	Risk	Severity	Workaround	Check Item
Modify the configuration item **spark.eventLog.dir**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Modify the configuration item **SPARK_DAEMON_JAVA_OPTS**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Delete all JobHistory2x instances.	The event logs of historical applications are lost.	▲ ▲	Reserve at least one JobHistory2x instance.	Check whether historical application information is included in JobHistory2x.
Delete or modify the **/user/spark2x/jars/8.1.0.1/spark-archive-2x.zip** file in HDFS.	JDBCServer2x fails to be started and service functions are abnormal.	▲ ▲ ▲	Delete **/user/spark2x/jars/8.1.0.1/spark-archive-2x.zip**, and wait for 10-15 minutes until the .zip package is automatically restored.	Check whether services can be started properly.

Storm High-Risk Operations

Table 14-16 Storm high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Modify the following plug-in related configuration items: • storm.scheduler • nimbus.authorizer • storm.thrift.transport • nimbus.blobstore.class • nimbus.topology.validator • storm.principal.tolocal	Services cannot start properly.	▲ ▲ ▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that the class names exist and are valid.	Check whether services can be started properly.

Operation	Risk	Severity	Workaround	Check Item
Modify the Storm instance **GC_OPTS** startup parameters, including: NIMBUS_GC_OPTS SUPERVISOR_GC_OPTS UI_GC_OPTS LOGVIEWER_GC_OPTS	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.
Modify the user resource pool configuration parameter **resource.aware.scheduler.user.pools**.	Services cannot run properly.	▲ ▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that resources allocated to each user are appropriate and valid.	Check whether services can be started and run properly
Change data directories.	If this operation is not properly performed, services may be abnormal and unavailable.	▲ ▲ ▲ ▲	Do not manually change data directories.	Check whether data directories are normal.
Restart services or instances.	The service will be interrupted for a short period of time, and ongoing operations will be interrupted.	▲ ▲ ▲	Restart services or instances when necessary.	Check whether the service is running properly and whether interrupted operations are restored.

Operation	Risk	Severity	Workaround	Check Item
Synchronize configurations (by restarting the required service).	The service will be restarted, resulting in temporary service interruption. If Supervisor is restarted, ongoing operations will be interrupted for a short period of time.	▲ ▲ ▲	Modify configuration when necessary.	Check whether the service is running properly and whether interrupted operations are restored.
Stop services or instances.	The service will be stopped, and related operations will be interrupted.	▲ ▲ ▲	Stop services when necessary.	Check whether the services are properly stopped.
Delete or modify metadata.	If Nimbus metadata is deleted, services are abnormal and ongoing operations are lost.	▲ ▲ ▲ ▲ ▲	Do not manually delete Nimbus metadata files.	Check whether Nimbus metadata files are normal.
Modify file permissions.	If permissions on the metadata and log directories are incorrectly modified, service exceptions may occur.	▲ ▲ ▲ ▲	Do not manually modify file permissions.	Check whether the permissions on the data and log directories are correct.
Delete topologies.	Topologies in use will be deleted.	▲ ▲ ▲ ▲	Delete topologies when necessary.	Check whether the topologies are successfully deleted.

Yarn High-Risk Operations

Table 14-17 Yarn high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Delete or change data directories **yarn.nodemanager.local-dirs** and **yarn.nodemanager.log-dirs**	This operation may cause service information loss.	▲ ▲ ▲	Do not delete data directories manually.	Check whether data directories are normal.

ZooKeeper High-Risk Operations

Table 14-18 ZooKeeper high-risk operations

Operation	Risk	Severity	Workaround	Check Item
Delete or change ZooKeeper data directories.	This operation may cause service information loss.	▲ ▲ ▲	Follow the capacity expansion guide to change the ZooKeeper data directories.	Check whether services and associated components are started properly.
Modify the ZooKeeper instance start parameter **GC_OPTS**.	Services cannot start properly.	▲ ▲	Strictly follow the prompt information when modifying related configuration items. Ensure that new values are valid.	Check whether services can be started properly.

Operation	Risk	Severity	Workaround	Check Item
Modify the znode ACL information in ZooKeeper.	If znode permission is modified in ZooKeeper, other users may have no permission to access the znode and some system functions are abnormal.	▲ ▲ ▲ ▲	During the modification, strictly follow the ZooKeeper Configuration Guide and ensure that other components can use ZooKeeper properly after ACL information modification.	Check that other components that depend on ZooKeeper can properly start and provide services.

15 FAQs

15.1 MRS Overview

15.1.1 What Is MRS Used For?

MapReduce Service (MRS) is an enterprise-grade big data platform that allows you to quickly build and operate economical, secure, full-stack, cloud-native big data environments on the cloud. It provides engines such as ClickHouse, Spark, Flink, Kafka, and HBase, and supports convergence of data lake, data warehouse, business intelligence (BI), and artificial intelligence (AI). Fully compatible with open-source components, MRS helps you rapidly innovate and expand service growth.

15.1.2 What Types of Distributed Storage Does MRS Support?

MRS supports Hadoop 3.1.*x* and will soon support other mainstream Hadoop versions released by the community. **Table 15-1** lists the component versions supported by MRS.

Table 15-1 MRS component versions

Component	MRS 1.9.2 (Applicable to MRS 1.9.*x*)	MRS 3.1.0
Alluxio	2.0.1	N/A
CarbonData	1.6.1	2.0.1
DBService	1.0.0	2.7.0
Flink	1.7.0	1.12.0
Flume	1.6.0	1.9.0
HBase	1.3.1	2.2.3
HDFS	2.8.3	3.1.1

Component	MRS 1.9.2 (Applicable to MRS 1.9.*x*)	MRS 3.1.0
Hive	2.3.3	3.1.0
Hudi	N/A	0.7.0
Hue	3.11.0	4.7.0
Impala	N/A	3.4.0
Kafka	1.1.0	2.11-2.4.0
KafkaManager	1.3.3.1	N/A
KrbServer	1.15.2	1.17
Kudu	N/A	1.12.1
LdapServer	1.0.0	2.7.0
Loader	2.0.0	N/A
MapReduce	2.8.3	3.1.1
Oozie	N/A	5.1.0
Opentsdb	2.3.0	N/A
Presto	0.216	333
Phoenix (integrated with HBase)	N/A	5.0.0
Ranger	1.0.1	2.0.0
Spark	2.2.2	N/A
Spark2x	N/A	2.4.5
Sqoop	N/A	1.4.7
Storm	1.2.1	N/A
Tez	0.9.1	0.9.2
YARN	2.8.3	3.1.1
ZooKeeper	3.5.1	3.5.6
MRS Manager	1.9.2	N/A
FusionInsight Manager	N/A	8.1.0

15.1.3 How Do I Create an MRS Cluster Using a Custom Security Group?

If you want to use a self-defined security group when buying a cluster, you need to enable port 9022 or select **Auto create** in **Security Group** on the MRS console.

15.1.4 How Do I Use MRS?

MapReduce Service (MRS) is a service you can use to deploy and manage Hadoop-based components on the Cloud. It enables you to deploy Hadoop clusters with a few clicks. MRS provides enterprise-ready big data clusters in the cloud. Tenants can fully control the clusters and easily run big data components such as Hadoop, Spark, HBase, Kafka, and Storm in the clusters.

MRS is easy to use. You can execute various tasks and process or store PB-scale data using computers connected in a cluster. To use MRS, do as follows:

1. Upload local programs and data files to OBS.
2. Create a cluster. You need to specify the cluster type (for example, analysis or streaming), and set ECS instance specifications, number of instances, data disk type (common I/O, high I/O, and ultra-high I/O), and components to be installed, such as Hadoop, Spark, HBase, Hive, Kafka, and Storm, in a cluster. You can use a bootstrap action to install third-party software or modify the cluster running environment on a node before or after the cluster is started.
3. Use MRS to submit, execute, and monitor your programs.
4. Manage clusters on MRS Manager, an enterprise-level unified management platform of big data clusters. You can learn about the health status of services and hosts, obtain critical system information in a timely manner from graphical metric monitoring and customization, modify service attributes based on performance requirements, and start or stop clusters, services, and role instances.
5. Terminate any MRS cluster that you do not require after job execution is complete.

15.1.5 How Does MRS Ensure Security of Data and Services?

MRS is a big data management and analytic platform featuring high security. It ensures data and service security from the following aspects:

- Network isolation

  The public cloud network is divided into service plane and management plane. The two planes are physically isolated to ensure network security.

  - Service plane: provides a network plane for running cluster components. The service plane provides service channels, and implements data access and storage, job submission, and computing.
  - Management plane: provides a public cloud management console that you can use to purchase and manage MRS.

- Host security

  You can deploy third-party antivirus software based on service requirements. MRS provides the following measures to improve security of OSs and ports:

  - OS kernel security hardening
  - OS patch update
  - OS permission control
  - OS port management
  - OS protocol and port attack defense
- Data security

  MRS enables data storage on OBS, thereby ensuring data security.
- Data integrity

  MRS transmits the processed data to OBS using SSL, thereby ensuring data integrity.

15.1.6 Can I Configure a Phoenix Connection Pool?

Phoenix does not support connection pool configuration. You are advised to write code to implement a tool class for managing connections and simulate a connection pool.

15.1.7 Does MRS Support Change of the Network Segment?

You can change the network segment. On the cluster **Dashboard** page of MRS console, click **Change Subnet** to the right of **Default Subnet**, and select a subnet in the VPC of the cluster to expand subnet IP addresses. Selecting a new subnet will not change the IP addresses and subnets of existing nodes.

15.1.8 Can I Downgrade the Specifications of an MRS Cluster Node?

You cannot downgrade the specifications of an MRS cluster node by using the console. If you want to downgrade an MRS cluster node's specifications, contact technical support.

15.1.9 What Is the Relationship Between Hive and Other Components?

- Hive and HDFS

  Hive is an Apache Hadoop project. Hive uses Hadoop Distributed File System (HDFS) as its file storage system. Hive parses and processes structured data stored on HDFS. All data files in the Hive database are stored in HDFS, and all data operations on Hive are also performed using HDFS APIs.
- Hive and MapReduce

  All data computing of Hive depends on MapReduce. MapReduce, also an Apache Hadoop project, is a parallel computing framework based on HDFS. During data analysis, Hive parses HiveQL statements submitted by users into MapReduce tasks and submits the tasks for MapReduce to execute.
- Hive and DBService

  MetaStore (metadata service) of Hive processes the structure and attribute information about Hive databases, tables, and partitions that are stored in a relational database. In MRS, the relational database is maintained by DBService.

- Hive and Spark

  Hive data computing can also be implemented on Spark. Spark, also an Apache project, is an in-memory distributed computing framework. During data analysis, Hive parses HiveQL statements submitted by users into Spark tasks and submits the tasks for Spark to execute.

15.1.10 Does an MRS Cluster Support Hive on Spark?

- Clusters of MRS 1.9.*x* support Hive on Spark.
- Clusters of MRS 3.*x* or later support Hive on Spark.
- You can use Hive on Tez for the clusters of other versions.

15.1.11 What Are the Differences Between Hive Versions?

Hive 3.1 has the following differences when compared with Hive 1.2:

- String cannot be converted to int.
- The user-defined functions (UDFs) of the **Date** type are changed to Hive built-in UDFs.
- Hive 3.1 does not provide the index function anymore.
- Hive 3.1 uses the UTC time in time functions, while Hive 1.2 uses the local time zone.
- The JDBC drivers in Hive 3.1 and Hive 1.2 are incompatible.
- In Hive 3.1, column names in ORC files are case-sensitive and underscores-sensitive.
- Hive 3.1 does not allow columns named **time**.

15.1.12 Which MRS Cluster Version Supports Hive Connection and User Synchronization?

MRS cluster 2.0.5 or later supports Hive connections on DataLake Governance Center (DGC) and provides the IAM user synchronization function.

15.1.13 What Are the Differences Between OBS and HDFS in Data Storage?

The data processed by MRS is from OBS or HDFS. OBS is an object-based storage service that provides secure, reliable, and cost-effective storage of huge amounts of data. MRS can directly process data in OBS. You can view, manage, and use data by using the OBS console or OBS client. In addition, you can use REST APIs independently or integrate APIs to service applications to manage and access data.

- Data stored in OBS: Data storage is decoupled from compute. The cluster storage cost is low, and storage capacity is not limited. Clusters can be deleted at any time. However, the computing performance depends on the OBS access performance and is lower than that of HDFS. OBS is recommended for applications that do not demand a lot of computation.
- Data stored in HDFS: Data storage is not decoupled from compute. The cluster storage cost is high, and storage capacity is limited. The computing performance is high. You must export data before you delete clusters. HDFS is recommended for computing-intensive scenarios.

15.1.14 How Do I Obtain the Hadoop Pressure Test Tool?

Download it from **https://github.com/Intel-bigdata/HiBench**.

15.1.15 What Is the Relationship Between Impala and Other Components?

- Impala and HDFS

  Impala uses HDFS as its file storage system. Impala parses and processes structured data, while HDFS provides reliable underlying storage. Impala provides fast data access without moving data in HDFS.

- Impala and Hive

  Impala uses Hive metadata, Open Database Connectivity (ODBC) driver, and SQL syntax. Unlike Hive, which is over MapReduce, Impala implements a distributed architecture based on daemon and handles all query executions on the same node. Therefore, Impala is faster than Hive by reducing the latency caused by MapReduce.

- Impala and MapReduce

  None

- Impala and Spark

  None

- Impala and Kudu

  Kudu can be closely integrated with Impala to replace the combination of Impala, HDFS, and Parquet. You can insert, query, update, and delete data in Kudu tablets using Impala's SQL syntax. In addition, you can use JDBC or ODBC to connect to Kudu for data operations, using Impala as the broker.

- Impala and HBase

  The default Impala tables use data files stored in HDFS, which is ideal for batch loading and query of full table scanning. However, HBase provides convenient and efficient query of OLTP-style organization data.

15.1.16 Statement About the Public IP Addresses in the Open-Source Third-Party SDK Integrated by MRS

The open-source third-party packages on which the open-source components integrated by MRS depend contain SDK usage examples. Public IP addresses such as 12.1.2.3, 54.123.4.56, 203.0.113.0, and 203.0.113.12 are example IP addresses. MRS will not initiate a connection to the public IP address or exchange data with the public IP address.

15.1.17 What Is the Relationship Between Kudu and HBase?

Kudu is designed based on the HBase structure and can implement fast random read/write and update functions that HBase is good at. Kudu and HBase are similar in architecture. The differences are as follows:

- HBase uses ZooKeeper to ensure data consistency, whereas Kudu uses the Raft consensus algorithm to ensure consistency.

- HBase uses HDFS for resilient data storage, whereas Kudu uses TServer to ensure strong data consistency and reliability.

15.1.18 Does MRS Support Running Hive on Kudu?

MRS does not support Hive on Kudu.

Currently, MRS supports only the following two methods to access Kudu:

- Access Kudu through Impala tables.
- Access and operate Kudu tables using the client application.

15.1.19 What Are the Solutions for processing 1 Billion Data Records?

- GaussDB (for MySQL) is recommended for scenarios, such as data updates, online transaction processing (OLTP), and complex analysis of 1 billion data records.
- Impala and Kudu in MRS also meet this requirement. Impala and Kudu can load all join tables to the memory in the join operation.

15.1.20 Can I Change the IP address of DBService?

MRS does not support the change of the DBService IP address.

15.1.21 Can I Clear MRS sudo Logs?

MRS sudo log files record operations performed by user **omm** and are helpful for fault locating. You can delete the logs of the earliest date to release storage space.

1. If the log file is large, add the log file directory to **/etc/logrotate.d/syslog** to enable the system to periodically delete logs.

   Method: Run **sed -i '3 a/var/log/sudo/sudo.log' /etc/logrotate.d/syslog**.

2. Set the maximum number and size of logs in **/etc/logrotate.d/syslog**. If the number or size of logs exceeds the threshold, the logs will be automatically deleted. By default, logs are aged based on the size and number of archived logs. You can use **size** and **rotate** to limit the size and number of archived logs, respectively. If required, you can also add **daily**/**weekly**/**monthly** to specify how often the logs are cleared.

15.1.22 Is the Storm Log also limited to 20 GB in MRS cluster 2.1.0?

In MRS cluster 2.1.0, the Storm log cannot exceed 20 GB. If the Storm log exceeds 20 GB, the log files will be deleted cyclically. Logs are stored on the system disk, therefore, the log space is limited. If you want to keep the log for longer time, mount the log directory to storage media.

15.1.23 What Is Spark ThriftServer?

ThriftServer is a JDBC API. You can use JDBC to connect to ThriftServer to access SparkSQL data. Therefore, you can see JDBCServer in Spark components, but not ThriftServer.

15.1.24 What Access Protocols Are Supported by Kafka?

Kafka supports PLAINTEXT, SSL, SASL_PLAINTEXT, and SASL_SSL.

15.1.25 What Is the Compression Ratio of zstd?

Zstandard (zstd) is an open-source fast lossless compression algorithm. The compression ratio of zstd is twice that of orc. For details, see **https://github.com/L-Angel/compress-demo**. CarbonData does not support lzo, and MRS has zstd integrated.

15.1.26 Why Are the HDFS, YARN, and MapReduce Components Unavailable When an MRS Cluster Is Created?

The HDFS, YARN, and MapReduce components are integrated in Hadoop. If the three components are unavailable when are MRS cluster is created, select Hadoop instead. After an MRS cluster is created, HDFS, YARN, and MapReduce are available in the **Components** page.

15.1.27 Why Is the ZooKeeper Component Unavailable When an MRS Cluster Is Created?

If you create a cluster of a version earlier than MRS 3.*x*, ZooKeeper is installed by default and is not displayed on the GUI.

If you create a cluster of MRS 3.*x* or later, ZooKeeper is available on the GUI and is selected by default.

After the cluster is created, the ZooKeeper component is available on the **Components** page.

15.1.28 Which Python Versions Are Supported by Spark Tasks in an MRS 3.1.0 Cluster?

For MRS 3.1.0 clusters, Python 2.7 or 3.*x* is recommended for Spark tasks.

15.1.29 How Do I Enable Different Service Programs to Use Different YARN Queues?

Create a tenant on Manager.

Procedure

Step 1 Log in to FusionInsight Manager and choose **Tenant Resources**.

Step 2 In the tenant list on the left, select a parent tenant and click  . On the page for adding a sub-tenant, set attributes for the sub-tenant according to **Table 15-2**.

Table 15-2 Sub-tenant parameters

Parameter	Description
Cluster	Indicates the cluster to which the parent tenant belongs.
Parent Tenant Resource	Indicates the name of the parent tenant.
Name	• Indicates the name of the current tenant. The value consists of 3 to 50 characters, including digits, letters, and underscores (_). • Plan a sub-tenant name based on service requirements. The name cannot be the same as that of a role, HDFS directory, or Yarn queue that exists in the current cluster.
Tenant Type	Specifies whether the tenant is a leaf tenant. • When **Leaf Tenant** is selected, the current tenant is a leaf tenant and no sub-tenant can be added. • When **Non-leaf Tenant** is selected, the current tenant is not a leaf tenant and sub-tenants can be added to the current tenant. However, the tenant depth cannot exceed 5 levels.
Computing Resource	Specifies the dynamic computing resources for the current tenant. • When **Yarn** is selected, the system automatically creates a queue in Yarn and the queue is named the same as the sub-tenant name. – A leaf tenant can directly submit jobs to the queue. – A non-leaf tenant cannot directly submit jobs to the queue. However, Yarn adds an extra queue (hidden) named **default** for the non-leaf tenant to record the remaining resource capacity of the tenant. Actual jobs do not run in this queue. • If **Yarn** is not selected, the system does not automatically create a queue.
Default Resource Pool Capacity (%)	Indicates the percentage of computing resources used by the current tenant. The base value is the total resources of the parent tenant.
Default Resource Pool Max Capacity (%)	Indicates the maximum percentage of computing resources used by the current tenant. The base value is the total resources of the parent tenant.

Parameter	Description
Storage Resource	Specifies storage resources for the current tenant. • When **HDFS** is selected, the system automatically creates a folder named after the sub-tenant in the HDFS parent tenant directory. • When **HDFS** is not selected, the system does not automatically allocate storage resources.
Quota	Indicates the quota for files and directories.
Space Quota	Indicates the quota for the HDFS storage space used by the current tenant. • If the unit is set to **MB**, the value ranges from **1** to **8796093022208**. If the unit is set to **GB**, the value ranges from **1** to **8589934592**. • This parameter indicates the maximum HDFS storage space that can be used by the tenant, but not the actual space used. • If its value is greater than the size of the HDFS physical disk, the maximum space available is the full space of the HDFS physical disk. • If this quota is greater than the quota of the parent tenant, the actual storage space does not exceed the quota of the parent tenant.
Storage Path	Indicates the HDFS storage directory for the tenant. • The system automatically creates a folder named after the sub-tenant name in the directory of the parent tenant by default. For example, if the sub-tenant is **ta1s** and the parent directory is **/tenant/ta1**, the storage path for the sub-tenant is then **/tenant/ta1/ta1s**. • The storage path is customizable in the parent directory.
Description	Indicates the description of the current tenant.

NOTE

Roles, computing resources, and storage resources are automatically created when tenants are created.

- The new role has permissions on the computing and storage resources. This role and its permissions are automatically controlled by the system and cannot be manually managed by choosing **System** > **Permission** > **Role**. The role name is in the format of *Tenant name_Cluster ID*. The ID of the first cluster is not displayed by default.
- When using this tenant, create a system user and bind the user to the role of the tenant.
- The sub-tenant can further allocate the resources of its parent tenant. The sum of the resource percentages of direct sub-tenants under a parent tenant at each level cannot exceed 100%. The sum of the computing resource percentages of all level-1 tenants cannot exceed 100%.

Step 3 Check whether the current tenant needs to be associated with resources of other services.

- If yes, go to **Step 4**.
- If no, go to **Step 5**.

Step 4 Click **Associate Service** to configure other service resources used by the current tenant.

1. Set **Services** to **HBase**.
2. Set **Association Type** as follows:
   - **Exclusive** indicates that the service resources are used by the tenant exclusively and cannot be associated with other tenants.
   - **Shared** indicates that the service resources can be shared with other tenants.

NOTE

- Only HBase can be associated with a new tenant. However, HDFS, HBase, and Yarn can be associated with existing tenants.
- To associate an existing tenant with service resources, click the target tenant in the tenant list, switch to the **Service Associations** page, and click **Associate Service** to configure resources to be associated with the tenant.
- To disassociate an existing tenant from service resources, click the target tenant in the tenant list, switch to the **Service Associations** page, and click **Delete** in the **Operation** column. In the displayed dialog box, select **I have read the information and understand the impact** and click **OK**.

3. Click **OK**.

Step 5 Click **OK**. Wait until the system displays a message indicating that the tenant is successfully created.

----End

15.1.30 Differences and Relationships Between the MRS Management Console and Cluster Manager

You can access Manager from the MRS management console.

Manager is classified as MRS Manager and FusionInsight Manager.

- MRS Manager is the manager page of MRS 2.*x* or earlier clusters.
- FusionInsight Manager is the manager page of MRS 3.*x* or later clusters.

The following table lists the differences and relationships between the management console and FusionInsight Manager.

Common Operation	MRS Console	FusionInsight Manager
Changing subnets, adding security group rules, controlling OBS permissions, managing agencies, and synchronizing IAM users	Supported	Not supported

Common Operation	MRS Console	FusionInsight Manager
Adding node groups, scaling out, scaling in, and upgrading specifications	Supported	Not supported
Isolating hosts, starting all roles, and stopping all roles	Supported	Supported
Downloading the client, starting services, stopping services, and perform rolling restart of services	Supported	Supported
Viewing the instance status of services, configuring parameters, and synchronizing configurations	Supported	Supported
Viewing cleared alarms and events	Supported	Supported
Viewing the alarm help	Not supported	Supported
Setting thresholds	Not supported	Supported
Adding message subscription specifications	Supported	Not supported
Managing files	Supported	Not supported
Managing jobs	Supported	Not supported
Managing tenants	Supported	Supported
Managing tags	Supported	Not supported
Managing permissions (adding and deleting users, user groups, and roles, and changing passwords)	Not supported	Supported
Performing backup and restoration	Not supported	Supported
Auditing	Not supported	Supported
Monitoring resources and logging	Supported	Supported

15.1.31 How Do I Unbind an EIP from an MRS Cluster Node?

Symptom

After an EIP is bound on the console, the EIP cannot be unbound in the EIP module of the VPC service.

A dialog box is displayed, indicating that the operation cannot be performed because the EIP is being used by MapReduce.

Procedure

Step 1 Log in to the VPC console and choose **Virtual Private Cloud** > **My VPCs**. Find the target VPC in the VPC list.

Step 2 Click the VPC name to go to the **Summary** tab page and click the number next to **Subnets** in the **Networking Components** area to find the subnet to which the cluster belongs.

Step 3 In the subnet list, click the target subnet name. Click the **IP Addresses** tab, locate the target public IP address and click **Unbind from EIP** in the **Operation** column.

----End

15.2 Account and Password

15.2.1 What Is the Account for Logging In to Manager?

The default account for logging in to Manager is **admin**, and the password is the one you set when you created the cluster.

15.2.2 How Do I Query and Change the Password Validity Period of an Account?

Querying the Password Validity Period

Querying the password validity period of a component running user (human-machine user or machine-machine user):

Step 1 Log in to the node where the client is installed as the client installation user.

Step 2 Run the following command to switch to the client directory, for example, **/opt/Bigdata/client**:

cd /opt/Bigdata/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command and enter the password of user **kadmin/admin** to log in to the kadmin console:

kadmin -p kadmin/admin

NOTE

The default password of user **kadmin/admin** is **Admin@123**. Change the password upon your first login or as prompted and keep the new password secure.

Step 5 Run the following command to view the user information:

getprinc *Internal system username*

Example: **getprinc user1**

```
kadmin:  getprinc user1
......
Expiration date: [never]
Last password change: Sun Oct 09 15:29:54 CST 2022
Password expiration date: [never]
......
```

----End

Querying the password validity period of an OS user:

Step 1 Log in to any master node in the cluster as user **root**.

Step 2 Run the following command to view the password validity period (value of **Password expires**):

chage -l *Username*

For example, to view the password validity period of user **root**, run the **chage -l root** command. The command output is as follows:

```
[root@xxx ~]#chage -l root
Last password change                                    : Sep 12, 2021
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
```

----End

Changing the Password Validity Period

- The password of a machine-machine user is randomly generated and never expires by default.
- The password validity period of a human-machine user can be changed by modifying the password policy on Manager.

15.3 Accounts and Permissions

15.3.1 Does an MRS Cluster Support Access Permission Control If Kerberos Authentication Is not Enabled?

For MRS cluster 2.1.0 or earlier, choose **System** > **Configuration** > **Permission** on MRS Manager.

For MRS cluster 3.*x* or later, choose **System** > **Permission** on FusionInsight Manager.

15.3.2 How Do I Assign Tenant Management Permission to a New Account?

You can assign tenant management permission only in analysis or hybrid clusters, but not in streaming clusters.

The operations vary depending on the MRS cluster version:

Procedure for versions earlier than MRS cluster 3.x:

Step 1 Log in to MRS Manager as user **admin**.

Step 2 Choose **System** > **Manage User**. Select the new account, and click **Modify** in the **Operation** column.

Step 3 In **Assign Rights by Role**, click **Select and Add Role**.

- If you bind the **Manager_tenant** role to the account, the account will have permission to view tenant management information.
- If you bind the **Manager_administrator** role to the account, the account will have permission to view and perform tenant management.

Step 4 Click **OK**.

----End

Procedure for MRS cluster 3.x and later versions:

Step 1 Log in to FusionInsight Manager and choose **System** > **Permission** > **User**.

Step 2 Locate the user and click **Modify**.

Modify the parameters based on service requirements.

If you bind the **Manager_tenant** role to the account, the account will have permission to view tenant management information. If you bind the **Manager_administrator** role to the account, the account will have permission to perform tenant management and view related information.

NOTE

It takes about three minutes for the settings to take effect after user group or role permission are modified.

Step 3 Click **OK**.

----End

15.3.3 How Do I Customize an MRS Policy?

1. On the IAM console, choose **Permissions** in the navigation pane, and click **Create Custom Policy**.
2. Set a policy name in **Policy Name**.
3. Set **Scope** to **Project-level service** for MRS.
4. Specify **Policy View**. The following options are supported:
   - **Visual editor**: Select cloud services, actions, resources, and request conditions from the navigation pane to customize the policy. You do not require knowledge of JSON syntax.
   - **JSON**: Edit JSON policies from scratch or based on an existing policy.

   You can also click **Select Existing Policy/Role** in the **Policy Content** area to select an existing policy as the template for modification.
5. (Optional) Enter a brief description in the **Description** area.
6. Click **OK**.
7. Attach the policy to a user group. Users in the group then inherit the permissions defined in this policy.

15.3.4 Why Is the Manage User Function Unavailable on the System Page on MRS Manager?

Check whether you have the **Manager_administrator** permission. If you do not have this permission, **Manage User** will not be available on the **System** page of MRS Manager.

15.3.5 Does Hue Support Account Permission Configuration?

Hue does not provide an entry for configuring account permissions on its web UI. However, you can configure user roles and user groups for Hue accounts on the **System** tab on Manager.

15.4 Client Usage

15.4.1 How Do I Configure Environment Variables and Run Commands on a Component Client?

1. Log in to any Master node as user **root**.
2. Run the **su - omm** command to switch to user **omm**.
3. Run the **cd /opt/client** command to switch to the client.
4. Run the **source bigdata_env** command to configure environment variables.

   If Kerberos authentication is enabled for the current cluster, run the **kinit** *Component service user* command to authenticate the user. If Kerberos authentication is disabled, skip this step.
5. After the environment variables are configured, run the client command of the component. For example, to view component information, you can run the HDFS client command **hdfs dfs -ls /** to view the HDFS root directory file.

15.4.2 How Do I Disable ZooKeeper SASL Authentication?

Log in to FusionInsight Manager, choose **Cluster** > **Services** > **ZooKeeper**, click the **Configurations** tab and then **All Configurations**. In the navigation pane on the left, choose **quorumpeer(Role)** > **Customization**, add the **set zookeeper.sasl.disable** parameter, and set its value to **false**. Save the configuration and restart the ZooKeeper service.

15.4.3 An Error Is Reported When the kinit Command Is Executed on a Client Node Outside an MRS Cluster

Symptom

After the client is installed on a node outside an MRS cluster and the **kinit** command is executed, the following error information is displayed:

```
-bash kinit Permission denied
```

The following error information is displayed when the **java** command is executed:

```
-bash: /xxx/java: Permission denied
```

After running the **ll /***Java installation path***/JDK/jdk/bin/java** command, it is found that the file execution permission is correct.

Fault Locating

Run the **mount | column -t** command to check the status of the mounted partition. It is found that the partition status of the mount point where the Java execution file is located is **noexec**. In the current environment, the data disk where the MRS client is installed is set to **noexec**, that is, binary file execution is prohibited. As a result, Java commands cannot be executed.

Solution

1. Log in to the node where the MRS client is located as user **root**.
2. Remove the configuration item **noexec** of the data disk where the MRS client is located from the **/etc/fstab** file.
3. Run the **umount** command to detach the data disk, and then run the **mount -a** command to remount the data disk.

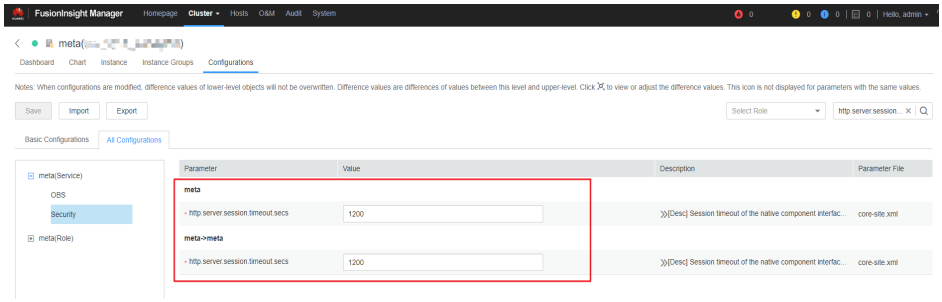
15.5 Web Page Access

15.5.1 How Do I Change the Session Timeout Duration for an Open Source Component Web UI?

You need to set a proper web session timeout duration for security purposes. To change the session timeout duration, do as follows:

Checking Whether the Cluster Supports Session Timeout Duration Adjustment

- For MRS cluster versions earlier than 3.x:
  a. On the cluster details page, choose **Components** > **meta** > **Service Configuration**.
  b. Switch **Basic** to **All**, and search for the **http.server.session.timeout.secs**.
     If **http.server.session.timeout.secs** does not exist, the cluster does not support change of the session timeout duration. If the parameter exists, perform the following steps to modify it.
- MRS 3.x and later: Log in to FusionInsight Manager and choose **Cluster** > **Services** > **meta**. On the displayed page, click **Configurations** and select **All Configurations**. Search for the **http.server.session.timeout.secs** configuration item. If this configuration item exists, perform the following steps to modify it. If the configuration item does not exist, the version does not support dynamic adjustment of the session duration.



You are advised to set all session timeout durations to the same value. Otherwise, the settings of some parameters may not take effect due to value conflict.

Modifying the Timeout Duration on Manager and the Authentication Center Page

For clusters of versions earlier than MRS 3.*x*:

1. Log in to each master node in the cluster and perform 2 to 4.
2. Change the value of **<session-timeout>20</session-timeout>** in the **/opt/Bigdata/apache-tomcat-7.0.78/webapps/cas/WEB-INF/web.xml** file. **<session-timeout>20</session-timeout>** indicates the session timeout duration, in minutes. Change it based on service requirements. The maximum value is 480 minutes.
3. Change the value of **<session-timeout>20</session-timeout>** in the **/opt/Bigdata/apache-tomcat-7.0.78/webapps/web/WEB-INF/web.xml** file. **<session-timeout>20</session-timeout>** indicates the session timeout duration, in minutes. Change it based on service requirements. The maximum value is 480 minutes.
4. Change the values of **p:maxTimeToLiveInSeconds="${tgt.maxTimeToLiveInSeconds:1200}"** and **p:timeToKillInSeconds="${tgt.timeToKillInSeconds:1200}"** in the **/opt/Bigdata/apache-tomcat-7.0.78/webapps/cas/WEB-INF/spring-configuration/ticketExpirationPolicies.xml** file. The maximum value is 28,800 seconds.

5. Restart the Tomcat node on the active master node.
   a. On the active master node, run the **netstat -anp |grep 28443 |grep LISTEN | awk '{print $7}'** command as user **omm** to query the Tomcat process ID.
   b. Run the **kill -9** *{pid}* command, in which *{pid}* indicates the Tomcat process ID obtained in **5.a**.
   c. Wait until the process automatically restarts. You can run the **netstat -anp |grep 28443 |grep LISTEN** command to check whether the process is successfully restarted. If the process can be queried, the process is successfully restarted. If the process cannot be queried, query the process again later.

For clusters of MRS 3.*x* or later

1. Log in to each master node in the cluster and perform **2** to **3** on each master node.
2. Change the value of **<session-timeout>20</session-timeout>** in the **/opt/Bigdata/om-server_xxx/apache-tomcat-xxx/webapps/web/WEB-INF/web.xml** file. **<session-timeout>20</session-timeout>** indicates the session timeout duration, in minutes. Change it based on service requirements. The maximum value is 480 minutes.
3. Add **ticket.tgt.timeToKillInSeconds=28800** to the **/opt/Bigdata/om-server_xxx/apache-tomcat-8.5.63/webapps/cas/WEB-INF/classes/config/application.properties** file. **ticket.tgt.timeToKillInSeconds** indicates the validity period of the authentication center, in seconds. Change it based on service requirements. The maximum value is 28,800 seconds.
4. Restart the Tomcat node on the active master node.
   a. On the active master node, run the **netstat -anp |grep 28443 |grep LISTEN | awk '{print $7}'** command as user **omm** to query the Tomcat process ID.
   b. Run the **kill -9** *{pid}* command, in which *{pid}* indicates the Tomcat process ID obtained in **4.a**.
   c. Wait until the process automatically restarts.

      You can run the **netstat -anp |grep 28443 |grep LISTEN** command to check whether the process is successfully restarted. If the process is displayed, the process is successfully restarted. If the process is not displayed, query the process again later.

Modifying the Timeout Duration for an Open-Source Component Web UI

1. Access the **All Configurations** page.
   - For MRS cluster versions earlier than MRS 3.x:

     On the cluster details page, choose **Components > Meta > Service Configuration**.
   - For MRS cluster version 3.*x* or later:

     Log in to FusionInsight Manager and choose **Cluster** > **Services** > **meta**. On the displayed page, click **Configurations** and select **All Configurations**.
2. Change the value of **http.server.session.timeout.secs** under **meta** as required. The unit is second.

3. Save the settings, deselect **Restart the affected services or instances**, and click **OK**.

   You are advised to perform the restart during off-peak hours.

4. (Optional) If you need to use the Spark web UI, search for **spark.session.maxAge** on the **All Configurations** page of Spark and change the value (in seconds).

   Save the settings, deselect **Restart the affected services or instances**, and click **OK**.

5. Restart the meta service and components on web UI, or restart the cluster during off-peak hours.

   To prevent service interruption, restart the service during off-peak hours or perform a rolling restart.

15.5.2 Why Cannot I Refresh the Dynamic Resource Plan Page on MRS Tenant Tab?

Step 1 Log in to the Master1 and Master2 nodes as user **root**.

Step 2 Run the **ps -ef |grep aos** command to check the AOS process ID.

Step 3 Run the **kill -9** *AOS process ID* command to end the AOS process.

Step 4 Wait until the AOS process is automatically restarted.

You can run the **ps -ef |grep aos** command to check whether the AOS process restarts successfully. If the process exists, the restart is successful and the **Dynamic Resource Plan** page will be refreshed. If the process does not exist, retry later.

----End

15.5.3 What Do I Do If the Kafka Topic Monitoring Tab Is Unavailable on Manager?

Step 1 Log in to each Master node of the cluster and switch to user **omm**.

Step 2 Go to the **/opt/Bigdata/apache-tomcat-7.0.78/webapps/web/WEB-INF/lib/components/Kafka/** directory.

Step 3 Run the **cp /opt/share/zookeeper-3.5.1-mrs-2.0/zookeeper-3.5.1-mrs-2.0.jar ./** command to copy the ZooKeeper package.

Step 4 Restart the Tomcat process.

sh /opt/Bigdata/apache-tomcat-7.0.78/bin/shutdown.sh

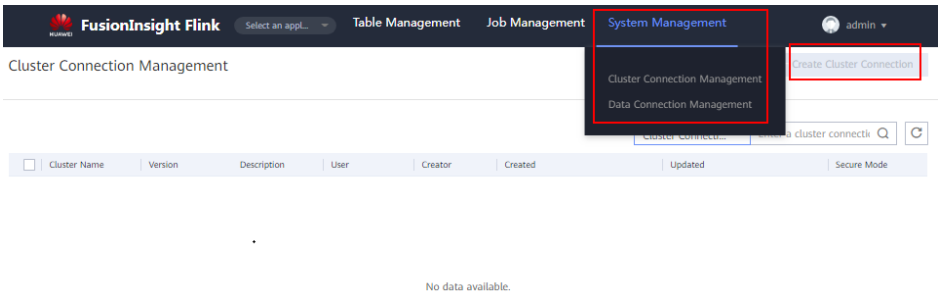
sh /opt/Bigdata/apache-tomcat-7.0.78/bin/startup.sh

----End

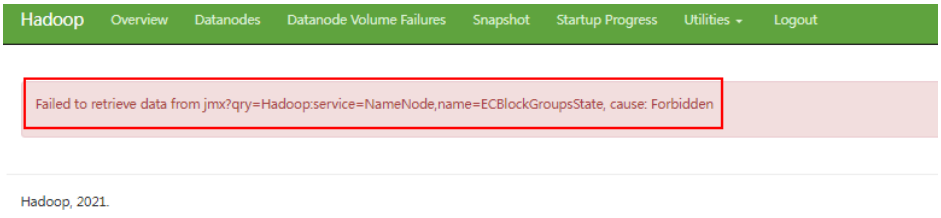
15.5.4 How Do I Do If an Error Is Reported or Some Functions Are Unavailable When I Access the Web UIs of HDFS, Hue, YARN, and Flink?

Users who access the web UIs of components such as HDFS, Hue, YARN, and Flink do not have required management permissions. As a result, an error is reported or some functions are unavailable. The following are some examples:

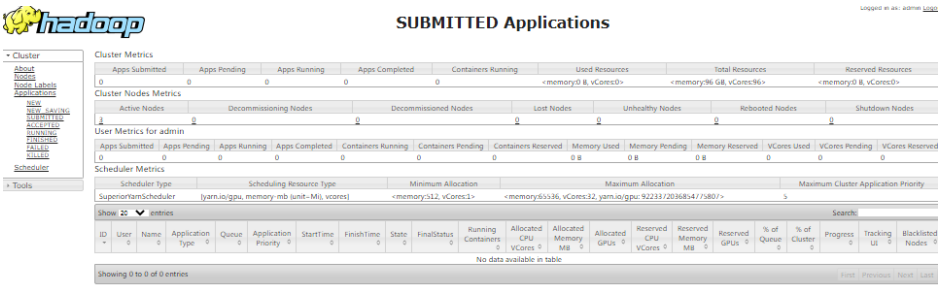
- After you log in to the web UI of Flink as the current user, some content cannot be displayed, and you do not have the permission to create applications, cluster connections, or data connections.



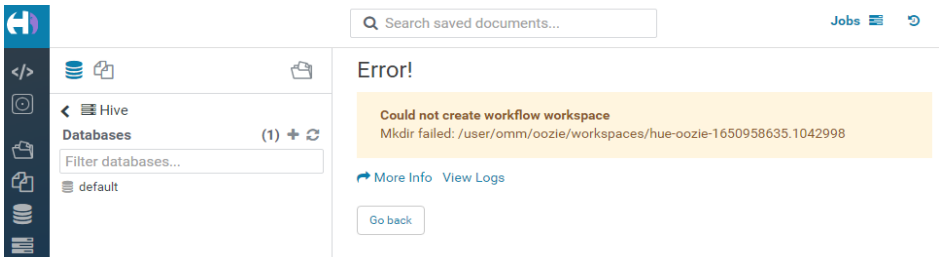
- After you log in to the web UI of HDFS as the current user, error message "Failed to retrieve data from /jmx?qry=java.lang:type=Memory, cause: Forbidden" is displayed.



- After you log in to the web UI of YARN as the current user, you cannot view job information.



- After you log in to the web UI of Hue as the current user, click  in the navigation pane on the left, and select **Workflow**, an error message is displayed.



You are advised to log in to the web UIs of the components as a user with corresponding management permissions. For example, you can create a service user who has the management permissions on HDFS and you can log in to the web UI of HDFS as the created user.

15.6 Alarm Monitoring

15.6.1 In an MRS Streaming Cluster, Can the Kafka Topic Monitoring Function Send Alarm Notifications?

The Kafka topic monitoring function cannot send alarms by email or SMS message. However, you can view alarm information on Manager.

15.6.2 Where Can I View the Running Resource Queues When the Alarm "ALM-18022 Insufficient Yarn Queue Resources" Is Reported?

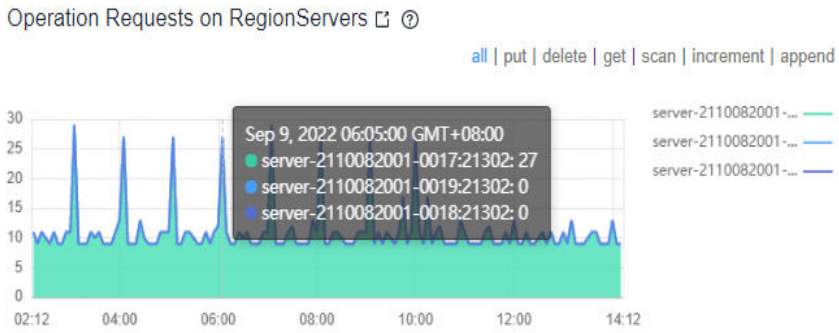
Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Yarn**. In the navigation pane on the left, choose **ResourceManager(Active)** and log in to the native Yarn page.

For details, see the online help.

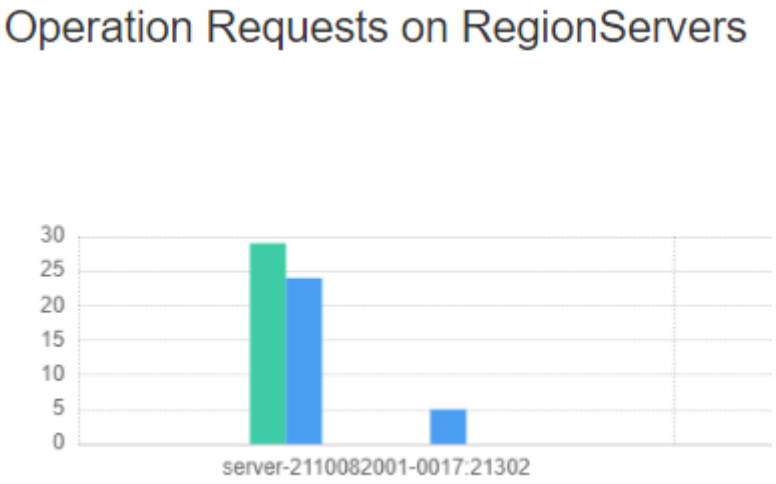
15.6.3 How Do I Understand the Multi-Level Chart Statistics in the HBase Operation Requests Metric?

The following uses the **Operation Requests on RegionServers** monitoring item as an example:

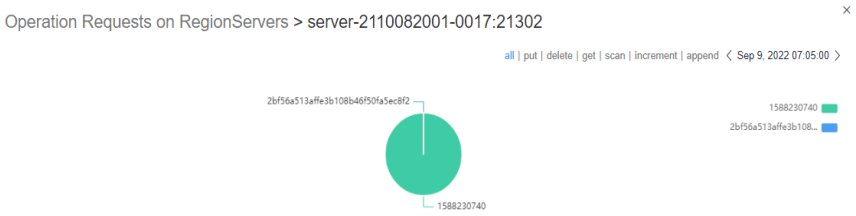
1. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **HBase** > **Resource**. On the displayed page, you can view the **Operation Requests on RegionServers** chart. If you click **all**, the top 10 RegionServers ranked by the total number of operation requests in the current cluster are displayed, the statistics interval is 5 minutes.



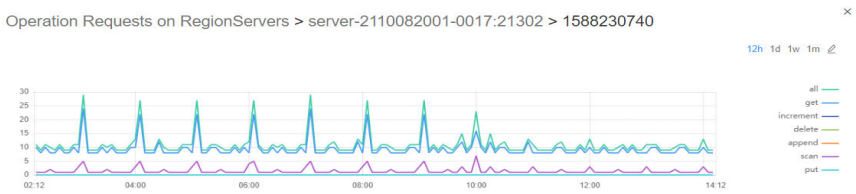
2. Click a point in the chart. A level-2 chart is displayed, showing the number of operation requests of all RegionServers in the past 5 minutes.



3. Click an operation statistics bar chart. A level-3 chart is displayed, showing the distribution of operations in each region within the period.



4. Click a region name. The distribution chart of operations performed every 5 minutes in the last 12 hours is displayed. You can view the number of operations performed in the period.



15.7 Performance Tuning

15.7.1 Does an MRS Cluster Support System Reinstallation?

An MRS cluster does not support system reinstallation.

15.7.2 Can I Change the OS of an MRS Cluster?

The OS of an MRS cluster cannot be changed.

15.7.3 How Do I Improve the Resource Utilization of Core Nodes in a Cluster?

1. Search for **yarn.nodemanager.resource.memory-mb**, and increase the value based on the actual memory of the cluster nodes.
2. Save the change and restart the affected services or instances.

15.7.4 How Do I Stop the Firewall Service?

Step 1 Log in to each node of a cluster as user **root**.

Step 2 Check whether the firewall service is started.

For example, to check the firewall status on EulerOS, run the **systemctl status firewalld.service** command.

Step 3 Stop the firewall service.

For example, to stop the firewall service on EulerOS, run the **systemctl stop firewalld.service** command.

----End

15.8 Job Development

15.8.1 How Do I Get My Data into OBS or HDFS?

MRS can process data in OBS and HDFS. You can get your data into OBS or HDFS as follows:

1. Upload local data to OBS.
   a. Log in to the OBS console.
   b. Create a parallel file system named **userdata** on OBS and create the **program**, **input**, **output**, and **log** folders in the file system.
      i. Choose **Parallel File System** > **Create Parallel File System**, and create a file system named **userdata**.
      ii. In the OBS file system list, click the file system name **userdata**, choose **Files** > **Create Folder**, and create the **program**, **input**, **output**, and **log** folders.
   c. Upload data to the **userdata** file system.
      i. Go to the **program** folder and click **Upload File**.

ii. Click **add file** and select a user program.

iii. Click **Upload**.

iv. Upload the user data file to the **input** directory using the same method.

2. Import OBS data to HDFS.

   You can import OBS data to HDFS only when **Kerberos Authentication** is disabled and the cluster is running.

   a. Log in to the MRS console.

   b. Click the name of the cluster.

   c. On the page displayed, select the **Files** tab page and click **HDFS File List**.

   d. Select a data directory, for example, **bd_app1**.

      The **bd_app1** directory is only an example. You can use any directory on the page or create a new one.

   e. Click **Import Data** and click **Browse** to select an OBS path and an HDFS path.

   f. Click **OK**.

      You can view the file upload progress on the **File Operation Records** tab page.

15.8.2 What Types of Spark Jobs Can Be Submitted in a Cluster?

MRS clusters support Spark jobs submitted in Spark, Spark Script, or Spark SQL mode.

15.8.3 Can I Run Multiple Spark Tasks at the Same Time After the Minimum Tenant Resources of an MRS Cluster Is Changed to 0?

You can run only one Spark task at a time after the minimum tenant resources of an MRS cluster is changed to 0.

15.8.4 What Are the Differences Between the Client Mode and Cluster Mode of Spark Jobs?

You need to understand the concept ApplicationMaster before understanding the essential differences between Yarn-client and Yarn-cluster.

In Yarn, each application instance has an ApplicationMaster process, which is the first container started by the application. It interacts with ResourceManager and requests resources. After obtaining resources, it instructs NodeManager to start containers. The essential difference between the Yarn-cluster and Yarn-client modes lies in the ApplicationMaster process.

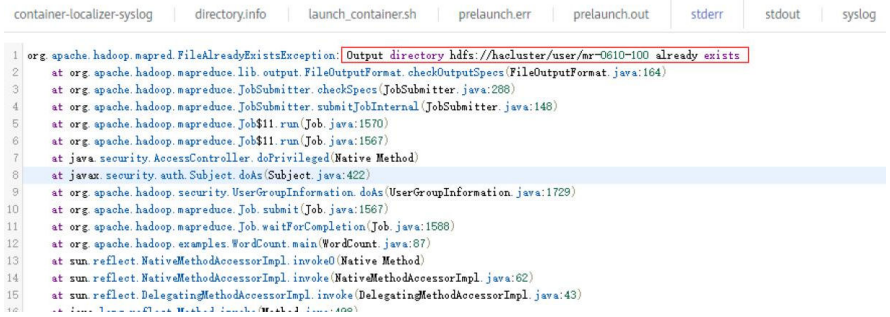
In Yarn-cluster mode, Driver runs in ApplicationMaster, which requests resources from Yarn and monitors the running status of a job. After a user submits a job, the client can be stopped and the job continues running on Yarn. Therefore, the Yarn-cluster mode is not suitable for running interactive jobs.

In Yarn-client mode, ApplicationMaster requests only Executor from Yarn. The client communicates with the requested containers to schedule tasks. Therefore, the client cannot be stopped.

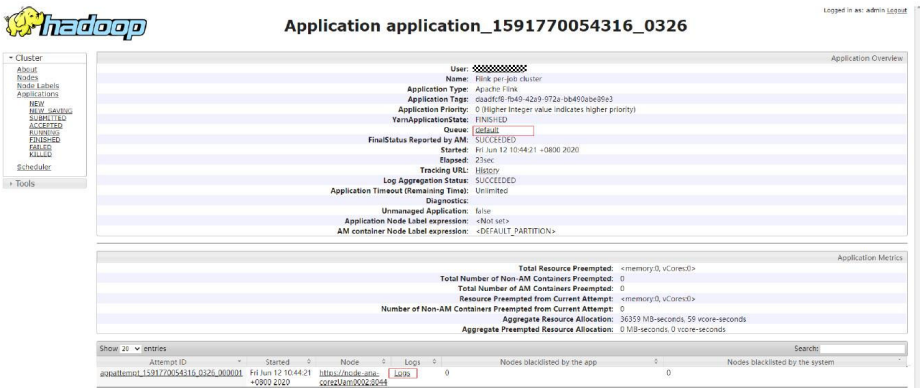
15.8.5 How Do I View MRS Job Logs?

Step 1 On the **Jobs** page of the MRS console, you can view logs of each job, including launcherJob and realJob logs.

- Generally, error logs are printed in **stderr** and **stdout** for launcherJob jobs, as shown in the following figure:



- You can view realJob logs on the ResourceManager web UI provided by the Yarn service on MRS Manager.



Step 2 Log in to the Master node of the cluster to obtain the job log files in **Step 1**. The HDFS path is **/tmp/logs/***{submit_user}*/**logs/***{application_id}*.

Step 3 After the job is submitted, if the job application ID cannot be found on the Yarn web UI, the job fails to be submitted. You can log in to the active Master node of the cluster and view the job submission process log **/var/log/executor/logs/exe.log**.

----End

15.8.6 How Do I Do If the Message "The current user does not exist on MRS Manager. Grant the user sufficient permissions on IAM and then perform IAM user synchronization on the Dashboard tab page." Is Displayed?

If IAM synchronization is not performed when a job is submitted in a security cluster, the error message "The current user does not exist on MRS Manager.

Grant the user sufficient permissions on IAM and then perform IAM user synchronization on the Dashboard tab page." is displayed.

Before submitting a job, on the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

15.8.7 LauncherJob Job Execution Is Failed And the Error Message "jobPropertiesMap is null." Is Displayed

The cause of the launcherJob failure is that the user who submits the job does not have the write permission on the **hdfs /mrs/job-properties** directory.

This problem is fixed in the 2.1.0.6 patch. You can also grant the write permission on the **/mrs/job-properties** directory to the synchronized user who submits the job on MRS Manager.

15.8.8 How Do I Do If the Flink Job Status on the MRS Console Is Inconsistent with That on Yarn?

To save storage space, the Yarn configuration item **yarn.resourcemanager.max-completed-applications** is modified to reduce the number of historical job records stored on Yarn. Flink jobs are long-term jobs. The realJob is still running on Yarn, but the launcherJob has been deleted. As a result, the launcherJob cannot be found on Yarn, and the job status fails to be updated. This problem is fixed in the 2.1.0.6 patch.

Workaround: Terminate the job whose launcherJob cannot be found. The status of the job submitted later will be updated.

15.8.9 How Do I Do If a SparkStreaming Job Fails After Being Executed Dozens of Hours and the OBS Access 403 Error is Reported?

When a user submits a job that needs to read and write OBS, the job submission program adds the temporary access key (AK) and secret key (SK) for accessing OBS by default. However, the temporary AK and SK have expiration time.

If you want to run long-term jobs such as Flink and SparkStreaming, you can enter the AK and SK in **Service Parameter** to ensure that the jobs will not fail to be executed due to key expiration.

15.8.10 How Do I Do If an Alarm Is Reported Indicating that the Memory Is Insufficient When I Execute a SQL Statement on the ClickHouse Client?

Symptom

The ClickHouse client restricts the memory used by GROUP BY statements. When a SQL statement is executed on the ClickHouse client, the following error information is displayed:

```
Progress: 1.83 billion rows, 85.31 GB (68.80 million rows/s., 3.21 GB/s.)  6%Received exception from server:
```

```
Code: 241. DB::Exception: Received from localhost:9000, 127.0.0.1.
DB::Exception: Memory limit (for query) exceeded: would use 9.31 GiB (attempt to allocate chunk of
1048576 bytes), maximum: 9.31 GiB:
(while reading column hits):
```

Solution

- Run the following command before executing an SQL statement on condition that the cluster has sufficient memory:
  ```
  SET max_memory_usage = 128000000000; #128G
  ```
- If no sufficient memory is available, ClickHouse enables you to overflow data to disk to free up the memory: You are advised to set the value of **max_memory_usage** to twice the size of **max_bytes_before_external_group_by**.
  ```
  set max_bytes_before_external_group_by=20000000000; #20G
  set max_memory_usage=40000000000; #40G
  ```

15.8.11 How Do I Do If Error Message "java.io.IOException: Connection reset by peer" Is Displayed During the Execution of a Spark Job?

Symptom

The Spark job keeps running and error message "java.io.IOException: Connection reset by peer" is displayed.

Solution

Add the **executor.memory Overhead** parameter to the parameters for submitting a job.

15.8.12 How Do I Do If Error Message "requestId=49718838510717737250" Is Displayed When a Spark Job Accesses OBS?

Symptom

Error message "requestId=4971883851071737250" is displayed when a Spark job accesses OBS.

Solution

Log in to the node where the Spark client is located, go to the **conf** directory, and change the value of the **fs.obs.metrics.switch** parameter in the **core-site.xml** configuration file to **false**.

15.8.13 Why DataArtsStudio Occasionally Fail to Schedule Spark Jobs and the Rescheduling also Fails?

Symptom

DataArtsStudio occasionally fails to schedule Spark jobs and the rescheduling also fails. The following error information is displayed:

```
Caused by: org.apache.spark.SparkException: Application application_1619511926396_2586346 finished with failed status
```

Solution

Log in to the node where the Spark client is located as user **root** and increase the value of the **spark.driver.memory** parameter in the **spark-defaults.conf** file.

15.8.14 How Do I Do If a Flink Job Fails to Execute and the Error Message "java.lang.NoSuchFieldError: SECURITY_SSL_ENCRYPT_ENABLED" Is Displayed?

Symptom

A Flink job fails to be executed and the following error message is displayed:

```
Caused by: java.lang.NoSuchFieldError: SECURITY_SSL_ENCRYPT_ENABLED
```

Solution

The third-party dependency package in the customer code conflicts with the cluster package. As a result, the job fails to be submitted to the MRS cluster. You need to modify the dependency package, set the scope of the open source Hadoop package and Flink package in the POM file to **provide**, and pack and execute the job again.

15.8.15 Why Submitted Yarn Job Cannot Be Viewed on the Web UI?

After a Yarn job is created, it cannot be viewed if you log in to the web UI as the **admin** user.

- The **admin** user is a user on the cluster management page. Check whether the user has the **supergroup** permission. Generally, only the user with the **supergroup** permission can view jobs.
- Log in to Yarn as the user who submits jobs to view jobs on Yarn. Do not view the jobs using the **admin** user.

15.8.16 How Do I Modify the HDFS NameSpace (fs.defaultFS) of an Existing Cluster?

You can modify or add the HDFS NameSpace (fs.defaultFS) of the cluster by modifying the **core-site.xml** and **hdfs-site.xml** files on the client. However, you are not advised to perform this operation on the server.

15.8.17 How Do I Do If the launcher-job Queue Is Stopped by YARN due to Insufficient Heap Size When I Submit a Flink Job on the Management Plane?

Symptom

The launcher-job queue is stopped by YARN when a Flink job is submitted on the management plane.

Solution

Increase the heap size of the launcher-job queue.

1. Log in to the active OMS node as user **omm**.
2. Change the value of **job.launcher.resource.memory.mb** in **/opt/executor/webapps/executor/WEB-INF/classes/servicebroker.xml** to **2048**.
3. Run the **sh /opt/executor/bin/restart-executor.sh** command to restart the executor process.

15.8.18 How Do I Do If the Error Message "slot request timeout" Is Displayed When I Submit a Flink Job?

Symptom

When a Flink job is submitted, JobManager is started successfully. However, TaskManager remains in the starting state until timeout. The following error information is displayed:

```
org.apache.flink.runtime.jobmanager.scheduler.NoResourceAvailableException: Could not allocate the required slot within slot request timeout. Please make sure that the cluster has enough resources
```

Possible Causes

1. The resources in the YARN queue are insufficient. As a result, TaskManager fails to start.
2. Your JAR files conflict with those in the environment. You can execute the WordCount program to determine whether the issue occurs.
3. If the cluster is in security mode, the SSL certificate of Flink may be incorrectly configured or has expired.

Solution

1. Add resources to the YARN queue.

2. Exclude the Flink and Hadoop dependencies in your JAR files so that Flink and Hadoop can depend only on the JAR files in the environment.
3. Reconfigure the SSL certificate of Flink..

15.8.19 Data Import and Export of DistCP Jobs

- Does a DistCP job compare data consistency during data import and export?

  No. DistCP jobs only copy data but do not modify it.
- When data is exported from a DistCP job, if some files already exist in OBS, how will the job process the files?

  DistCP jobs will overwrite the files in OBS.

15.9 Cluster Upgrade/Patching

15.9.1 Can I Upgrade an MRS Cluster?

You cannot upgrade an MRS cluster. However, you can create a cluster of the target version and migrate data from the old cluster to the new cluster.

15.9.2 Can I Change the MRS Cluster Version?

You cannot change the version of an MRS cluster. However, you can terminate the current cluster and create an MRS cluster of the version you require.

15.10 Cluster Access

15.10.1 Can I Switch Between the Two Login Modes of MRS?

No. You can select the login mode when creating the cluster. You cannot change the login mode after you created the cluster.

15.10.2 How Can I Obtain the IP Address and Port Number of a ZooKeeper Instance?

You can obtain the IP address and port number of a ZooKeeper instance through the MRS console or FusionInsight Manager.

Method 1: Obtaining the IP address and port number of a ZooKeeper through the MRS console

1. On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.
2. Click the **Components** tab and choose **ZooKeeper**. On the displayed page, click **Instances** to view the business IP address of a ZooKeeper instance.
3. Click the **Service Configuration** tab. On the displayed page, search for the **clientPort** parameter to view the port number of the ZooKeeper instance.

Method 2: Obtaining the IP address and port number of a ZooKeeper through FusionInsight Manager

1. Log in to FusionInsight Manager. For details, see .
2. Perform the following operations to obtain the IP address and port number of a ZooKeeper instance.
   - For clusters of MRS 3.*x* or earlier
     i. Choose **Services** > **ZooKeeper**. On the displayed page, click the **Instance** tab to view the business IP address of a ZooKeeper instance.
     ii. Click the **Service Configuration** tab. On the displayed page, search for the **clientPort** parameter to view the port number of the ZooKeeper instance.
   - For clusters of MRS 3.*x* or later
     i. Choose **Cluster** > **Services** > **ZooKeeper**. On the displayed page, click the **Instance** tab to view the business IP address of a ZooKeeper instance.
     ii. Click the **Configurations** tab. On the displayed page, search for the **clientPort** parameter to view the port number of the ZooKeeper instance.

15.10.3 How Do I Access an MRS Cluster from a Node Outside the Cluster?

Creating a Linux ECS Outside the Cluster to Access the MRS Cluster

Step 1 Create an ECS outside the cluster.

Set **AZ**, **VPC**, and **Security Group** of the ECS to the same values as those of the cluster to be accessed.

Step 2 On the VPC management console, apply for an EIP and bind it to the ECS.

Step 3 Configure security group rules for the cluster.

1. On the **Dashboard** tab page, click **Add Security Group Rule**. In the **Add Security Group Rule** dialog box that is displayed, click **Manage Security Group Rule**.
2. Click the **Inbound Rules** tab, and click **Add Rule**. In the **Add Inbound Rule** dialog box, configure the IP address of the ECS and enable all ports.
3. After the security group rule is added, you can download and install the client on the ECS..
4. Use the client.

   Log in to the client node as the client installation user and run the following command to switch to the client directory:

   cd /opt/hadoopclient

   Run the following command to load environment variables:

   source bigdata_env

   If Kerberos authentication is enabled for the cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, authentication is not required.

kinit *MRS cluster user*

Example:

kinit admin

Run the client command of a component.

Example:

Run the following command to view files in the HDFS root directory:

hdfs dfs -ls /

```
Found 15 items
drwxrwx--x   - hive       hive              0 2021-10-26 16:30 /apps
drwxr-xr-x   - hdfs       hadoop            0 2021-10-18 20:54 /datasets
drwxr-xr-x   - hdfs       hadoop            0 2021-10-18 20:54 /datastore
drwxrwx---+  - flink      hadoop            0 2021-10-18 21:10 /flink
drwxr-x---   - flume      hadoop            0 2021-10-18 20:54 /flume
drwxrwx--x   - hbase      hadoop            0 2021-10-30 07:31 /hbase
...
```

----End

15.11 Big Data Service Development

15.11.1 Can MRS Run Multiple Flume Tasks at a Time?

The Flume client supports multiple independent data flows. You can configure and link multiple sources, channels, and sinks in the **properties.properties** configuration file. These components can be linked to form multiple flows.

The following is an example of configuring two data flows in a configuration file:

```
server.sources = source1 source2
server.sinks = sink1 sink2
server.channels = channel1 channel2

#dataflow1
server.sources.source1.channels = channel1
server.sinks.sink1.channel = channel1

#dataflow2
server.sources.source2.channels = channel2
server.sinks.sink2.channel = channel2
```

15.11.2 How Do I Change FlumeClient Logs to Standard Logs?

1. Log in to the node where FlumeClient is running.
2. Go to the FlumeClient installation directory.

   For example, if the FlumeClient installation directory is **/opt/FlumeClient**, run the following command:

   cd /opt/FlumeClient/fusioninsight-flume-1.9.0/bin
3. Run the **./flume-manage.sh stop** command to stop FlumeClient.
4. Run the **vi /log4j.properties** command to open the **log4j.properties** file and change the value of **flume.root.logger** to **${flume.log.level},console**.
5. Run the **vim /flume-manager.sh** command to open the **flume-manager.sh** script in the **bin** directory in the Flume installation directory.

6. Comment out the following information in the **flume-manager.sh** script:

   >/dev/null 2>&1 &

7. Run the **./flume-manage.sh start** command to restart FlumeClient.
8. After the modification, check whether the Docker configuration is correct.

15.11.3 Where Are the .jar Files and Environment Variables of Hadoop Located?

- The **hadoopstreaming.jar** file is stored in the **/opt/share/hadoop-streaming-*** directory. ***** indicates the Hadoop version.
- The JDK environment variables are stored in **/opt/client/JDK/component_env**.
- The Hadoop environment variables are stored in **/opt/client/HDFS/component_env**.
- The Hadoop client path is **/opt/client/HDFS/hadoop**.

15.11.4 What Compression Algorithms Does HBase Support?

HBase supports the Snappy, LZ4, and gzip compression algorithms.

15.11.5 Can MRS Write Data to HBase Through the HBase External Table of Hive?

No. Hive on HBase supports only data query.

15.11.6 How Do I View HBase Logs?

1. Log in to the Master node in the cluster as user **root**.
2. Run the **su - omm** command to switch to user **omm**.
3. Run the **cd /var/log/Bigdata/hbase/** command to go to the **/var/log/Bigdata/hbase/** directory and view HBase logs.

15.11.7 How Do I Set the TTL for an HBase Table?

- Set the time to live (TTL) when creating a table:

  Create the **t_task_log** table, set the column family to **f**, and set the TTL to **86400** seconds.

  ```
  create 't_task_log',{NAME => 'f', TTL=>'86400'}
  ```

- Set the TTL for an existing table:

  ```
  disable "t_task_log" #Disable the table (services must be stopped).
  alter "t_task_log",NAME=>'data',TTL=>'86400' # Set the TTL value for the column family data.
  enable "t_task_log" #Restore the table.
  ```

15.11.8 How Do I Balance HDFS Data?

1. Log in to the master node of the cluster and run the corresponding command to configure environment variables. **/opt/client** indicates the client installation directory. Replace it with the actual one.

   source /opt/client/bigdata_env

kinit Component service user (If Kerberos authentication is enabled for the cluster, run this command to authenticate the user. Skip this step if the Kerberos authentication is disabled.)

2. Run the following command to start the balancer:

   /opt/client/HDFS/hadoop/sbin/start-balancer.sh -threshold 5

3. View the log.

   After you execute the balance task, the **hadoop-root-balancer-***Host name*.**log** log file will be generated in the client installation directory **/opt/client/HDFS/hadoop/logs**.

4. (Optional) If you do not want to perform data balancing, run the following commands to stop the balancer:

   source /opt/client/bigdata_env

   kinit Component service user (If Kerberos authentication is enabled for the cluster, run this command to authenticate the user. Skip this step if the Kerberos authentication is disabled.)

   /opt/client/HDFS/hadoop/sbin/stop-balancer.sh -threshold 5

15.11.9 How Do I Change the Number of HDFS Replicas?

1. Search for **dfs.replication**, change the value (value range: 1 to 16), and restart the HDFS instance.

15.11.10 What Is the Port for Accessing HDFS Using Python?

The default port of open source HDFS is **50070** for versions earlier than MRS 3.0.0, and **9870** for MRS 3.0.0 or later. **Common HDFS Ports** describes the common ports of HDFS.

Common HDFS Ports

The protocol type of all ports in the table is TCP.

Parameter	Default Port	Port Description
dfs.namenode.rpc.port	• 9820 (versions earlier than MRS 3.*x*) • 8020 (MRS 3.*x* and later)	NameNode RPC port This port is used for: 1. Communication between the HDFS client and NameNode 2. Connection between the DataNode and NameNode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
dfs.namenode.http.port	9870	HDFS HTTP port (NameNode) This port is used for: 1. Point-to-point NameNode checkpoint operations. 2. Connecting the remote web client to the NameNode UI **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.namenode.https.port	9871	HDFS HTTPS port (NameNode) This port is used for: 1. Point-to-point NameNode checkpoint operations 2. Connecting the remote web client to the NameNode UI **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.datanode.ipc.port	9867	IPC server port of DataNode This port is used for: Connection between the client and DataNode to perform RPC operations. **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
dfs.datanode.port	9866	DataNode data transmission port This port is used for: 1. Transmitting data from HDFS client from or to the DataNode 2. Point-to-point DataNode data transmission **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.datanode.http.port	9864	DataNode HTTP port This port is used for: Connecting to the DataNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.datanode.https.port	9865	HTTPS port of DataNode This port is used for: Connecting to the DataNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
dfs.JournalNode.rpc.port	8485	RPC port of JournalNode This port is used for: Client communication to access multiple types of information **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.journalnode.http.port	8480	JournalNode HTTP port This port is used for: Connecting to the JournalNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes
dfs.journalnode.https.port	8481	HTTPS port of JournalNode This port is used for: Connecting to the JournalNode from the remote web client in security mode **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. ● Is the port enabled by default during the installation: Yes ● Is the port enabled after security hardening: Yes

Parameter	Default Port	Port Description
httpfs.http.port	14000	Listening port of the HttpFS HTTP server This port is used for: Connecting to the HttpFS from the remote REST API **NOTE** The port ID is a recommended value and is specified based on the product. The port range is not restricted in the code. • Is the port enabled by default during the installation: Yes • Is the port enabled after security hardening: Yes

15.11.11 How Do I Modify the HDFS Active/Standby Switchover Class?

If the **org.apache.hadoop.hdfs.server.namenode.ha.AdaptiveFailoverProxyProvider** class is unavailable when a cluster of MRS 3.*x* connects to NameNodes using HDFS, the cause is that the HDFS active/standby switchover class of the cluster is configured improperly. To solve the problem, perform the following operations:

- Method 1: Add the **hadoop-plugins-*xxx*.jar** package to the **classpath** or **lib** directory of your program.

  The **hadoop-plugins-*xxx*.jar** package is stored in the HDFS client directory, for example, **$HADOOP_HOME/share/hadoop/common/lib/hadoop-plugins-8.0.2-302023.jar**.
- Method 2: Change the configuration item of HDFS to the corresponding open source class, as shown in the follows:

  dfs.client.failover.proxy.provider.hacluster=org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider

15.11.12 What Is the Recommended Number Type of DynamoDB in Hive Tables?

smallint is recommended.

15.11.13 Can the Hive Driver Be Interconnected with DBCP2?

The Hive driver cannot be interconnected with the DBCP2 database connection pool. The DBCP2 database connection pool invokes the **isValid** method to check whether a connection is available. However, Hive directly throws an exception when implementing this method.

15.11.14 How Do I View the Hive Table Created by Another User?

Versions earlier than MRS 3.*x*:

1. Log in to MRS Manager and choose **System** > **Permission** > **Manage Role**.
2. Click **Create Role**, and set **Role Name** and **Description**.
3. In the **Permission** table, choose **Hive** > **Hive Read Write Privileges**.
4. In the database list, click the name of the database where the table created by user B is stored. The table is displayed.
5. In the **Permission** column of the table created by user B, select **SELECT**.
6. Click **OK**, and return to the **Role** page.
7. Choose **System** > **Manage User**. Locate the row containing user A, click **Modify** to bind the new role to user A, and click **OK**. After about 5 minutes, user A can access the table created by user B.

MRS 3.*x* or later:

1. Log in to FusionInsight Manager and choose **Cluster** > **Services**. On the page that is displayed, choose **Hive**. On the displayed page, choose **More**, and check whether **Enable Ranger** is grayed out.
   - If yes, go to **9**.
   - If no, perform **2** to **8**.
2. Log in to FusionInsight Manager and choose **System** > **Permission** > **Role**.
3. Click **Create Role**, and set **Role Name** and **Description**.
4. In the **Configure Resource Permission** table, choose *Name of the desired cluster* > **Hive** > **Hive Read Write Privileges**.
5. In the database list, click the name of the database where the table created by user B is stored. The table is displayed.
6. In the **Permission** column of the table created by user B, select **Select**.
7. Click **OK**, and return to the **Role** page.
8. Choose **Permission** > **User**. On the **Local User** page that is displayed, locate the row containing user A, click **Modify** in the **Operation** column to bind the new role to user A, and click **OK**. After about 5 minutes, user A can access the table created by user B.
9. Perform the following steps to add the Ranger access permission policy of Hive:
   a. Log in to FusionInsight Manager as a Hive administrator and choose **Cluster** > **Services**. On the page that is displayed, choose **Ranger**. On the displayed page, click the URL next to **Ranger WebUI** to go to the Ranger management page.
   b. On the home page, click the component plug-in name in the **HADOOP SQL** area, for example, **Hive**.
   c. On the **Access** tab page, click **Add New Policy** to add a Hive permission control policy.
   d. In the **Create Policy** dialog box that is displayed, set the following parameters:

- **Policy Name**: Enter a policy name, for example, **table_test_hive**.
- **database**: Enter or select the database where the table created by user B is stored, for example, **default**.
- **table**: Enter or select the table created by user B, for example, **test**.
- **column**: Enter and select a column, for example, *****.
- In the **Allow Conditions** area, click **Select User**, select user A, click **Add Permissions**, and select **select**.
- Click **Add**.

10. Perform the following steps to add the Ranger access permission policy of HDFS:
    a. Log in to FusionInsight Manager as user **rangeradmin** and choose **Cluster** > **Services**. On the page that is displayed, choose **Ranger**. On the displayed page, click the URL next to **Ranger WebUI** to go to the Ranger management page.
    b. On the home page, click the component plug-in name in the **HDFS** area, for example, **hacluster**.
    c. Click **Add New Policy** to add a HDFS permission control policy.
    d. In the **Create Policy** dialog box that is displayed, set the following parameters:
        - **Policy Name**: Enter a policy name, for example, **tablehdfs_test**.
        - **Resource Path**: Set this parameter to the HDFS path where the table created by user B is stored, for example, **/user/hive/warehouse/***Database name***/***Table name*.
        - In the **Allow Conditions** area, select user A for **Select User**, click **Add Permissions** in the **Permissions** column, and select **Read** and **Execute**.
        - Click **Add**.
11. View basic information about the policy in the policy list. After the policy takes effect, user A can view the table created by user B.

15.11.15 Can I Export the Query Result of Hive Data?

Run the following statement to export the query result of Hive data:

```
insert overwrite local directory "/tmp/out/" row format delimited fields terminated by "\t" select * from table;
```

15.11.16 How Do I Do If an Error Occurs When Hive Runs the beeline -e Command to Execute Multiple Statements?

When Hive of MRS 3.*x* runs the **beeline -e " use default;show tables;"** command, the following error message is displayed: Error while compiling statement: FAILED: ParseException line 1:11 missing EOF at ';' near 'default' (state=42000,code=40000).

Solutions:

- Method 1: Replace the **beeline -e " use default;show tables;"** command with **beeline --entirelineascommand=false -e "use default;show tables;"**.
- Method 2:
  a. In the **/opt/Bigdata/client/Hive** directory on the Hive client, change **export CLIENT_HIVE_ENTIRELINEASCOMMAND=true** in the **component_env** file to **export CLIENT_HIVE_ENTIRELINEASCOMMAND=false**.

  Figure 15-1 Changing the **component_env** file

  b. Run the following command to verify the configuration:

  source /opt/Bigdata/client/bigdata_env

  beeline -e " use default;show tables;"

15.11.17 How Do I Do If a "hivesql/hivescript" Job Fails to Submit After Hive Is Added?

This issue occurs because the **MRS CommonOperations** permission bound to the user group to which the user who submits the job belongs does not include the Hive permission after being synchronized to Manager. To solve this issue, perform the following operations:

1. Add the Hive service.
2. Log in to the IAM console and create a user group. The policy bound to the user group is the same as that of the user group to which the user who submits the job belongs.
3. Add the user who submits the job to the new user group.
4. Refresh the cluster details page on the MRS console. The status of IAM user synchronization is **Not synchronized**.
5. Click **Synchronize** on the right of **IAM User Sync**. Go back to the previous page. In the navigation pane on the left, choose **Operation Logs** and check whether the user is changed.
   - If yes, submit the Hive job again.
   - If no, check whether all the preceding operations are complete.
     - If yes, contact the O&M personnel.
     - If no, submit the Hive job after the preceding operations are complete.

15.11.18 What If an Excel File Downloaded on Hue Failed to Open?

1. Log in to a Master node as user **root** and switch to user **omm**.

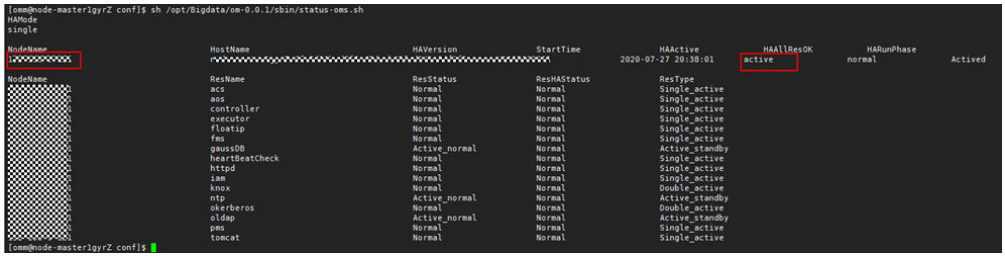
su - omm

2. Check whether the current node is the active OMS node.

   sh ${BIGDATA_HOME}/om-0.0.1/sbin/status-oms.sh

   If **active** is displayed in the command output, the node is the active node. Otherwise, log in to the other Master node.

Figure 15-2 Active OMS node



3. Go to the **${BIGDATA_HOME}/Apache-httpd-*/conf** directory.

   cd ${BIGDATA_HOME}/Apache-httpd-*/conf

4. Open the **httpd.conf** file.

   vim httpd.conf

5. Search for **21201** in the file and delete the following content from the file (The values of *proxy_ip* and *proxy_port* in **Figure 15-3** are examples only):

   ProxyHTMLEnable On
   SetEnv PROXY_PREFIX=https://[proxy_ip]:[proxy_port]
   ProxyHTMLURLMap (https?:\/\/[^:]*:[0-9]*.*) ${PROXY_PREFIX}/proxyRedirect=$1 RV

Figure 15-3 Content to be deleted

6. Save the modification and exit.
7. Open the **httpd.conf** file again, search for **proxy_hue_port**, and delete the following content:

   ProxyHTMLEnable On
   SetEnv PROXY_PREFIX=https://[proxy_ip]:[proxy_port]
   ProxyHTMLURLMap (https?:\/\/[^:]*:[0-9]*.*) ${PROXY_PREFIX}/proxyRedirect=$1 RV

Figure 15-4 Content to be deleted

8. Save the modification and exit.
9. Run the following command to restart the **httpd** process:

   sh ${BIGDATA_HOME}/Apache-httpd-*/setup/restarthttpd.sh
10. Check whether the **httpd.conf** file on the standby Master node is modified. If the file is modified, no further action is required. If the file is not modified, modify the **httpd.conf** file on the standby Master node in the same way. You do not need to restart the **httpd** process.
11. Download the Excel file again. You can open the file successfully.

15.11.19 How Do I Do If Sessions Are Not Released After Hue Connects to HiveServer and the Error Message "over max user connections" Is Displayed?

Applicable versions: MRS 3.1.0 and earlier

1. Modify the following file on the two Hue nodes:

   /opt/Bigdata/FusionInsight_Porter_8.*/install/FusionInsight-Hue-*/hue/apps/beeswax/src/beeswax/models.py
2. Change the configurations in lines 396 and 404.

   Change **q Changed = self.filter(owner=user, application=application).exclude(guid='').exclude(secret='')** to **q = self.filter(owner=user, application=application).exclude(guid=None).exclude(secret=None)**.

15.11.20 How Do I Reset Kafka Data?

You can reset Kafka data by deleting Kafka topics.

- Delete a topic: **kafka-topics.sh --delete --zookeeper** *ZooKeeper Cluster service IP address*:**2181/kafka --topic** ***topicname***
- Query all topics: **kafka-topics.sh --zookeeper** *ZooKeeper cluster service IP address*:**2181/kafka --list**

After the deletion command is executed, empty topics will be deleted immediately. If a topic has data, the topic will be marked for deletion and will be deleted by Kafka later.

15.11.21 How Do I Obtain the Client Version of MRS Kafka?

Run the **--bootstrap-server** command to query the information about the client.

15.11.22 What Access Protocols Are Supported by Kafka?

Kafka supports PLAINTEXT, SSL, SASL_PLAINTEXT, and SASL_SSL.

15.11.23 How Do I Do If Error Message "Not Authorized to access group xxx" Is Displayed When a Kafka Topic Is Consumed?

This issue is caused by the conflict between the Ranger authentication and ACL authentication of a cluster. If a Kafka cluster uses ACL for permission access control and Ranger authentication is enabled for the Kafka component, all authentications of the component are managed by Ranger. The permissions set by the original authentication plug-in are invalid. As a result, ACL authorization does not take effect. You can disable Ranger authentication of Kafka and restart the Kafka service to rectify the fault. The procedure is as follows:

1. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Kafka**.
2. In the upper right corner of the **Dashboard** page, click **More** and choose **Disable Ranger**. In the displayed dialog box, enter the password and click **OK**. After the operation is successful, click **Finish**.
3. In the upper right corner of the **Dashboard** page, click **More** and choose **Restart Service** to restart the Kafka service.

15.11.24 What Compression Algorithms Does Kudu Support?

Kudu supports **Snappy**, **LZ4**, and **zlib**. **LZ4** is used by default.

15.11.25 How Do I View Kudu Logs?

1. Log in to the Master node in the cluster.
2. Run the **su - omm** command to switch to user **omm**.
3. Run the **cd /var/log/Bigdata/kudu/** command to go to the **/var/log/Bigdata/kudu/** directory and view Kudu logs.

15.11.26 How Do I Handle the Kudu Service Exceptions Generated During Cluster Creation?

Viewing the Kudu Service Exception Logs

1. Log in to the MRS console.
2. Click the name of the cluster.
3. On the page displayed, choose **Components** > **Kudu** > **Instances** and locate the IP address of the abnormal instance.

   If the **Components** tab is unavailable, complete IAM user synchronization first. (On the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.)
4. Log in to the node where the abnormal instance resides, and view the Kudu log.

   ```
   cd /var/log/Bigdata/Kudu
   [root@node-master1AERu kudu]# ls
   healthchecklog  runninglog  startlog
   ```

   You can find the Kudu health check logs in the **healthchecklog** directory, the startup logs in the **startlog** directory, and the Kudu process run logs in the **runninglog** directory.

```
[root@node-master1AERu logs]# pwd
/var/log/Bigdata/kudu/runninglog/master/logs
[root@node-master1AERu logs]# ls -al
kudu-master.ERROR   kudu-master.INFO   kudu-master.WARNING
```

Run logs are classified into three types: ERROR, INFO, and WARNING. Each type of run logs is recorded in the corresponding file. You can run the **cat** command to view run logs of each type.

Handling Kudu Service Exceptions

The **/var/log/Bigdata/kudu/runninglog/master/logs/kudu-master.INFO** file contains the following error information:

```
"Unable to init master catalog manager: not found: Unable to initialize catalog manager: Failed to initialize sys tables async: Unable to load consensus metadata for tablet 00000000000000000000000000: xxx"
```

If this exception occurs when the Kudu service is installed for the first time, the KuduMaster service is not started. The data inconsistency causes the startup failure. To solve the problem, perform the following steps to clear the data directories and restart the Kudu service. If the Kudu service is not installed for the first time, clearing the data directories will cause data loss. In this case, migrate data and clear the data directory.

1. Search for the data directories **fs_data_dir**, **fs_wal_dir**, and **fs_meta_dir**.

   find /opt -name master.gflagfile

   cat /opt/Bigdata/FusionInsight_Kudu_*/*_KuduMaster/etc/master.gflagfile | grep fs_

2. On the cluster details page, choose **Components** > **Kudu** and click **Stop Service**.
3. Clear the Kudu data directories on all KuduMaster and KuduTserver nodes. The following command uses two data disks as an example.

   rm -Rvf /srv/Bigdata/data1/kudu, rm -Rvf /srv/Bigdata/data2/kudu

4. On the cluster details page, choose **Components** > **Kudu** and choose **More** > **Restart Service**.
5. Check the Kudu service status and logs.

15.11.27 Does OpenTSDB Support Python APIs?

OpenTSDB supports Python APIs. OpenTSDB provides HTTP-based RESTful APIs that are language-independent. Any language that supports HTTP requests can interconnect to OpenTSDB.

15.11.28 How Do I Configure Other Data Sources on Presto?

In this section, MySQL is used as an example.

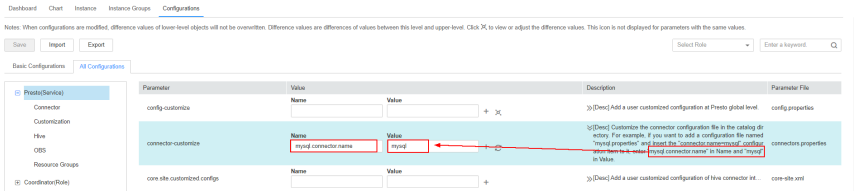
- For MRS 1.*x* and 3.*x* clusters, do the following:
  a. Log in to the MRS management console.
  b. Click the name of the cluster to go to its details page.
  c. Click the **Components** tab and then **Presto** in the component list. On the page that is displayed, click the **Configurations** tab then the **All Configurations** sub-tab.

d. On the Presto configuration page that is displayed, find **connector-customize**.

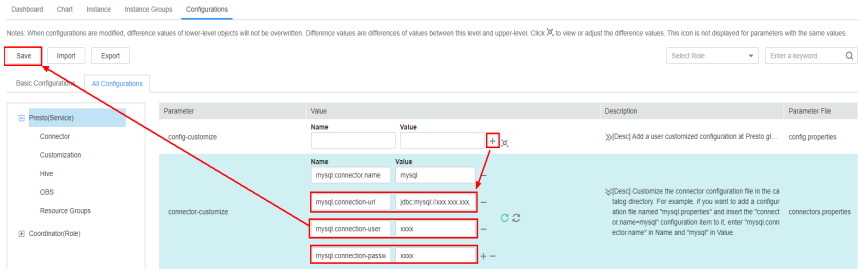
e. Set **Name** and **Value** as follows:

Name: **mysql.connector.name**

Value: **mysql**



f. Click the plus sign (+) to add three more fields and set **Name** and **Value** according to the table below. Then click **Save**.



Name	Value	Description
mysql.connection-url	jdbc:mysql://xxx.xxx.xxx.xxx:3306	Database connection pool
mysql.connection-user	xxxx	Database username
mysql.connection-password	xxxx	Database password

g. Restart the Presto service.

h. Run the following command to connect to the Presto Server of the cluster:

presto_cli.sh **--***krb5-config-path* {krb5.conf path} --*krb5-principal* {User principal} --*krb5-keytab-path* {user.keytab path} --*user* {presto username}

i. Log in to Presto and run the **show catalogs** command to check whether the data source list mysql of Presto can be queried.

```
[root@node-master2uoHG bin]# ./presto_cli.sh
--server http://10.[redacted]20
       show catalogs;
 Catalog
---------
 hive
 jmx
 mysql
 system
 tpcds
 tpch
(6 rows)

Query 20220422_121338_00002_ra2vb, FINISHED, 3 nodes
Splits: 53 total, 53 done (100.00%)
0:00 [0 rows, 0B] [0 rows/s, 0B/s]
```

Run the **show schemas from mysql** command to query the MySQL database.

- For MRS 2.*x* clusters, do the following:
  a. Create the **mysql.properties** configuration file containing the following content:

     connector.name=mysql

     connection-url=jdbc:mysql://mysqlIp:3306

     connection-user=Username

     connection-password=Password

     NOTE

     - **mysqlIp** indicates the IP address of the MySQL instance, which must be able to communicate with the MRS network.
     - The username and password are those used to log in to the MySQL database.

  b. Upload the configuration file to the **/opt/Bigdata/MRS_Current/1_14_Coordinator/etc/catalog/** directory on the master node (where the Coordinator instance resides) and the **/opt/Bigdata/MRS_Current/1_14_Worker/etc/catalog/** directory on the core node (depending on the actual directory in the cluster), and change the file owner group to **omm:wheel**.
  c. Restart the Presto service.

15.11.29 How Do I Connect to Spark Shell from MRS?

1. Log in to the Master node in the cluster as user **root**.
2. Run the following command to configure environment variables:

   source /opt/client/bigdata_env
3. If Kerberos authentication is enabled for the cluster, authenticate the user. If Kerberos authentication is disabled, skip this step.

   Command: **kinit** *MRS cluster user*

   Example:

   – If the user is a machine-machine user, run **kinit -kt user.keytab sparkuser**.
   – If the user is a human-machine user, run **kinit sparkuser**.
4. Run the following command to connect to Spark shell:

   spark-shell

15.11.30 How Do I Connect to Spark Beeline from MRS?

1. Log in to the master node in the cluster as user **root**.
2. Run the following command to configure environment variables:

   source /opt/client/bigdata_env
3. If Kerberos authentication is enabled for the cluster, authenticate the user. If Kerberos authentication is disabled, skip this step.

   Command: **kinit** *MRS cluster user*

   Example:

- If the user is a machine-machine user, run **kinit -kt user.keytab sparkuser**.
- If the user is a human-machine user, run **kinit sparkuser**.

4. Run the following command to connect to Spark Beeline:

   spark-beeline

5. Run commands on Spark Beeline. For example, create the table **test** in the **obs://mrs-word001/table/** directory.

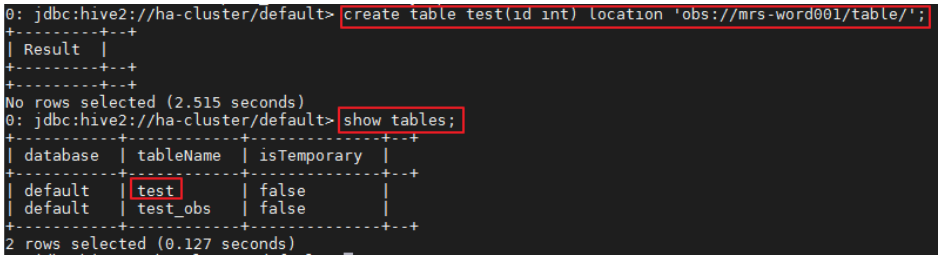
   create table test(id int) location 'obs://mrs-word001/table/';

6. Query all tables.

   show tables;

   If the table **test** is displayed in the command output, OBS is successfully accessed.

   Figure 15-5 Returned table name

7. Press **Ctrl+C** to exit the Spark Beeline.

15.11.31 Where Are the Execution Logs of Spark Jobs Stored?

- Logs of unfinished Spark jobs are stored in the **/srv/BigData/hadoop/data1/nm/containerlogs/** directory on the Core node.
- Logs of finished Spark jobs are stored in the **/tmp/logs/***username*/**logs** directory of HDFS.

15.11.32 How Do I Specify a Log Path When Submitting a Task in an MRS Storm Cluster?

You can modify the **/opt/Bigdata/MRS_***XXX* **/1_***XX* **_Supervisor/etc/worker.xml** file on the streaming Core node of MRS, set the value of **filename** to the path, and restart the corresponding instance on Manager.

You are advised not to modify the default log configuration of MRS. Otherwise, the log system may become abnormal.

15.11.33 How Do I Check Whether the ResourceManager Configuration of Yarn Is Correct?

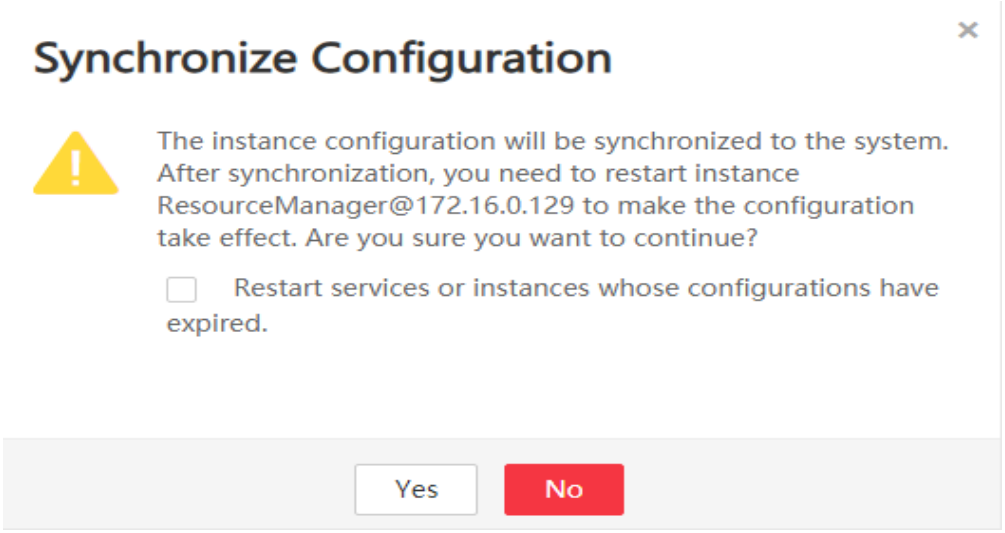
Step 1 Log in to MRS Manager and choose **Services** > **Yarn** > **Instance**.

Step 2 Synchronize the configuration between the two ResourceManager nodes.

Perform the following steps on each ResourceManager node:

1. Click the name of the ResourceManager node, and choose **More** > **Synchronize Configuration**.
2. In the dialog box displayed, deselect **Restart services or instances whose configurations have expired** and click **Yes**.

Figure 15-6 Synchronization configurations



Step 3 Log in to the Master nodes as user **root**.

Step 4 Run the **cd /opt/Bigdata/MRS_Current/*_*_ResourceManager/etc_UPDATED/** command to go to the **etc_UPDATED** directory.

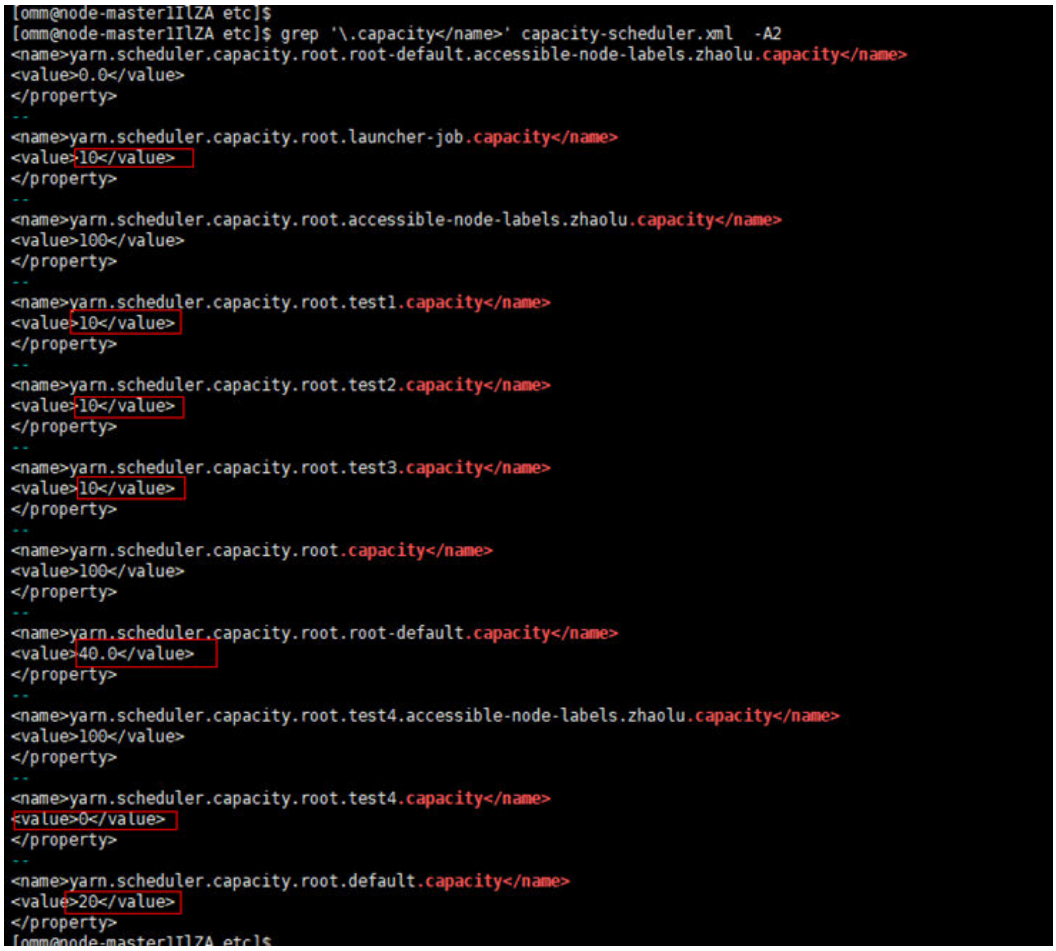
Step 5 Run the **grep '\.queues' capacity-scheduler.xml -A2** command to display all configured queues and check whether the queues are consistent with those displayed on Manager.

root-default is hidden on the Manager page.

```
[omm@node-master1IlZA etc]$
[omm@node-master1IlZA etc]$ grep '\.queues' capacity-scheduler.xml  -A2
<name>yarn.scheduler.capacity.root.queues</name>
<value>default,root-default,launcher-job,test1,test2,test3,test4</value>
</property>
[omm@node-master1IlZA etc]$
[omm@node-master1IlZA etc]$
```

Step 6 Run the **grep '\.capacity</name>' capacity-scheduler.xml -A2** command to display the value of each queue and check whether the value of each queue is the same as that displayed on Manager. Check whether the sum of the values configured for all queues is **100**.

- If the sum is **100**, the configuration is correct.
- If the sum is not **100**, the configuration is incorrect. Perform the following steps to rectify the fault.



Step 7 Log in to MRS Manager, and select **Hosts**.

Step 8 Determine the active Master node. The host name of the active Master node starts with a solid pentagon.

Step 9 Log in to the active Master node as user **root**.

Step 10 Run the **su - omm** command to switch to user **omm**.

Step 11 Run the **sh /opt/Bigdata/om-0.0.1/sbin/restart-controller.sh** command to restart the controller when no operation is being performed on Manager.

Restarting the controller will not affect the big data component services.

Step 12 Repeat **Step 1** to **Step 6** to synchronize ResourceManager configurations and check whether the configurations are correct.

If the latest configuration has not been loaded after the configuration synchronization is complete, a message will be displayed on the Manager page indicating that the configuration has expired. However, this will not affect services. The latest configuration will be automatically loaded when the component restarts.

----End

15.11.34 How Do I Modify the allow_drop_detached Parameter of ClickHouse?

Step 1 Log in to the node where the ClickHouse client is located as user **root**.

Step 2 Run the following commands to go to the client installation directory and set the environment variables:

cd /opt/*Client installation directory*

source bigdata_env

Step 3 If Kerberos authentication is enabled for the cluster, run the following command to authenticate the user. If Kerberos authentication is disabled, skip this step.

kinit *MRS cluster user*

NOTE

The user must have the ClickHouse administrator permissions.

Step 4 Run the **clickhouse client --host** *192.168.42.90* **--secure -m** command, in which *192.168.42.90* indicates the IP address of the ClickHouseServer instance node. The command output is as follows:

```
[root@server-2110082001-0017 hadoopclient]# clickhouse client --host 192.168.42.90 --secure -m
ClickHouse client version 21.3.4.25.
Connecting to 192.168.42.90:21427.
Connected to ClickHouse server version 21.3.4 revision 54447.
```

Step 5 Run the following command to set the value of the **allow_drop_detached** parameter, for example, **1**:

set allow_drop_detached=1;

Step 6 Run the following command to query the value of the **allow_drop_detached** parameter:

SELECT * FROM system.settings WHERE name = 'allow_drop_detached';

Step 7 Run the **q;** command to exit the ClickHouse client.

----End

15.11.35 How Do I Do If an Alarm Indicating Insufficient Memory Is Reported During Spark Task Execution?

Symptom

When a Spark task is executed, an alarm indicating insufficient memory is reported. The alarm ID is 18022. As a result, no available memory can be used.

Procedure

Set the executor parameters in the SQL script to limit the number of cores and memory of an executor.

For example, the configuration is as follows:

```
set hive.execution.engine=spark;
set spark.executor.cores=2;
set spark.executor.memory=4G;
set spark.executor.instances=10;
```

Change the values of the parameters as required.

15.11.36 How Do I Do If ClickHouse Consumes Excessive CPU Resources?

Symptom

A user performs a large number of update operations using ClickHouse. This operation on a ClickHouse consumes a large number of resources. In addition, the operation will be executed again if it fails. As a result, retries of those failed operations occupy too many CPU resources.

Procedure

Delete existing data from ZooKeeper and release delete the update statement.

15.11.37 How Do I Enable the Map Type on ClickHouse?

Step 1 Log in to the active Master node as user **root**.

Step 2 Run the following command to modify the **/opt/Bigdata/components/current/ClickHouse/configurations.xml** configuration file to enable user parameter customization:

vim /opt/Bigdata/components/current/ClickHouse/configurations.xml

Change **hidden** to **advanced**, as shown in the following information in bold. Then save the configuration and exit.

```
<property type="hidden" scope="all" classification="Customization"
classdesc="RESID_CLICKHOUSE_CONF_0056">
       <name>_clickhouse.custom_content.key</name>
       <value>_user-xml-content</value>
</property>
<property type="advanced" scope="all" classification="Customization"
classdesc="RESID_CLICKHOUSE_CONF_0056">
      <name>_user-xml-content</name>
      <value vType="text" checker="clickhouse.xmlformat">&lt;yandex&gt;&lt;/yandex&gt;</value>
      <description>RESID_CLICKHOUSE_CONF_0025</description>
</property>
```

Step 3 Run the following commands to switch to user **omm** and restart the controller service:

su - omm

sh /opt/Bigdata/om-server/om/sbin/restart-controller.sh

Step 4 Log in to FusionInsight Manager, choose **Cluster** > **Services** > **ClickHouse**. On the page that is displayed, click the **Configurations** tab then the **All Configurations** sub-tab. Click **ClickHouseServer(Role)** > **Customization**, and add the following content to the **_user-xml-content** configuration item in the right pane:

```
<yandex>
  <profiles>
    <default>
   <allow_experimental_map_type>1</allow_experimental_map_type>
    </default>
  </profiles>
</yandex>
```

Step 5 Click **Save**.

Step 6 Choose **Cluster** > **Services** > **ClickHouse**. In the upper right corner, choose **More** > **Restart Service** to restart the ClickHouse service.

----End

15.11.38 A Large Number of OBS APIs Are Called When Spark SQL Accesses Hive Partitioned Tables

Symptom

When Spark SQL is used to access Hive partitioned tables stored in OBS, the acces speed is slow and a large number of OBS query APIs are called.

Example SQL:

```
select a,b,c from test where b=xxx
```

Fault Locating

According to the configuration, the task should scan only the partition whose b is *xxx*. However, the task logs show that the task scans all partitions and then calculates the data whose b is *xxx*. As a result, the task calculation is slow. In addition, a large number of OBS requests are sent because all files need to be scanned.

By default, the execution plan optimization based on partition statistics is enabled on MRS, which is equivalent to automatic execution of Analyze Table. (The default configuration method is to set **spark.sql.statistics.fallBackToHdfs** to **true**. You can set this parameter to **false**.) After this function is enabled, table partition statistics are scanned during SQL execution and used as cost estimation in the execution plan. For example, small tables identified during cost evaluation are broadcast to each node in the memory for join operations, significantly reducing shuffle time. This function greatly optimizes performance in join scenarios, but increases the number of OBS calls.

Procedure

Set the following parameter in Spark SQL and then run the SQL statement:

```
set spark.sql.statistics.fallBackToHdfs=false;
```

Alternatively, run the **--conf** command to set this parameter to **false** before startup.

```
--conf spark.sql.statistics.fallBackToHdfs=false
```

15.12 API

15.12.1 How Do I Configure the node_id Parameter When Using the API for Adjusting Cluster Nodes?

When you use the API for adjusting cluster nodes, the value of **node_id** is fixed to **node_orderadd**.

15.13 Cluster Management

15.13.1 How Do I View All Clusters?

You can view all MRS clusters on the **Clusters** page. You can view clusters in different status.

- **Active Clusters**: all clusters except clusters in **Failed** and **Terminated** states.
- **Cluster History**: clusters in the **Terminated** state. Only the clusters terminated within the last six months are displayed. If you want to view clusters terminated more than six months ago, contact technical support engineers.
- **Failed Tasks**: tasks in **Failed** state. The failed tasks include the following:
  - Tasks failed to create clusters
  - Tasks failed to terminate clusters
  - Tasks failed to scale out clusters
  - Tasks failed to scale in clusters

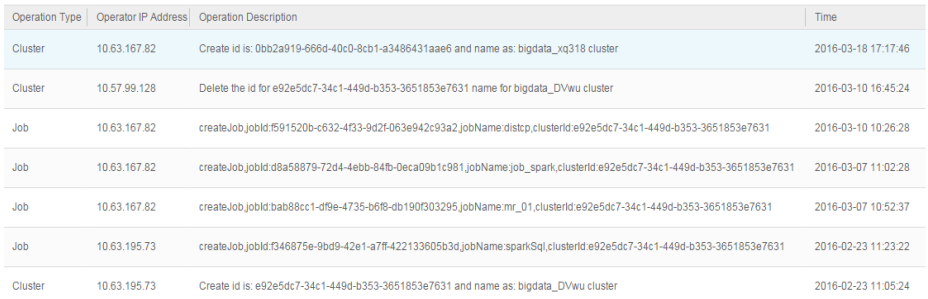
15.13.2 How Do I View Log Information?

You can view operation logs of clusters and jobs on the **Operation Logs** page. The MRS operation logs record the following operations:

- Cluster operations
  - Create, terminate, and scale out or in clusters
  - Create directories and delete directories or files
- Job operations: Create, stop, and delete jobs
- Data operations: IAM user tasks, add users, and add user groups

Figure 15-7 shows the operation logs.

Figure 15-7 Log information



Operation Type	Operator IP Address	Operation Description	Time
Cluster	10.63.167.82	Create id is: 0bb2a919-666d-40c0-8cb1-a3486431aae6 and name as: bigdata_xq318 cluster	2016-03-18 17:17:46
Cluster	10.57.99.128	Delete the id for e92e5dc7-34c1-449d-b353-3651853e7631 name for bigdata_DVwu cluster	2016-03-10 16:45:24
Job	10.63.167.82	createJob,jobId:f591520b-c632-4f33-9d2f-063e942c93a2,jobName:distcp,clusterId:e92e5dc7-34c1-449d-b353-3651853e7631	2016-03-10 10:26:28
Job	10.63.167.82	createJob,jobId:d8a58879-72d4-4ebb-84fb-0eca09b1c981,jobName:job_spark,clusterId:e92e5dc7-34c1-449d-b353-3651853e7631	2016-03-07 11:02:28
Job	10.63.167.82	createJob,jobId:bab88cc1-df9e-4735-b6f8-db190f303295,jobName:mr_01,clusterId:e92e5dc7-34c1-449d-b353-3651853e7631	2016-03-07 10:52:37
Job	10.63.195.73	createJob,jobId:f346875e-9bd9-42e1-a7ff-422133605b3d,jobName:sparkSql,clusterId:e92e5dc7-34c1-449d-b353-3651853e7631	2016-02-23 11:23:22
Cluster	10.63.195.73	Create id is: e92e5dc7-34c1-449d-b353-3651853e7631 and name as: bigdata_DVwu cluster	2016-02-23 11:05:24

15.13.3 How Do I View Cluster Configuration Information?

- After a cluster is created, click the cluster name on the MRS console. On the page displayed, you can view basic configuration information about the cluster. The instance specifications and node capacity determine the data analysis and processing capability. Higher instance specifications and larger capacity enable faster data processing at a higher cost.
- On the basic information page, click **Access Manager** to access the MRS cluster management page. On MRS Manager, you can view and handle alarms, and modify cluster configuration.

15.13.4 How Do I Install Kafka and Flume in an MRS Cluster?

You cannot install the Kafka and Flume components for a created cluster of MRS 3.1.0 or earlier. Kafka and Flume are components for a streaming cluster. To install Kafka and Flume, create a streaming or hybrid cluster, and install Kafka and Flume.

15.13.5 How Do I Stop an MRS Cluster?

To stop an MRS cluster, stop each node in the cluster on the ECS. Click the name of each node on the **Nodes** tab page to go to the **Elastic Cloud Server** page and click **Stop**.

15.13.6 Can I Expand Data Disk Capacity for MRS?

You can expand data disk capacity for MRS during off-peak hours.

Expand the EVS disk capacity, and then log in to the ECS and expand the partitions and file system. MRS nodes are installed using public images and support the capacity expansion of in-use EVS disks.

15.13.7 Can I Add Components to an Existing Cluster?

You cannot add or remove any component to and from a created cluster of MRS 3.1.0. However, you can create an MRS cluster that contains the required components.

15.13.8 Can I Delete Components Installed in an MRS Cluster?

You cannot delete any component from a created MRS cluster of MRS 3.1.0. If a component is not required, log in to MRS Manager and stop the component on the **Services** page.

15.13.9 Can I Change MRS Cluster Nodes on the MRS Console?

You cannot change MRS cluster nodes on the MRS console. You are also advised not to change MRS cluster nodes on the ECS console. Manually stopping or deleting an ECS, modifying or reinstalling the ECS OS, or modifying ECS specifications for a cluster node on the ECS console will affect the cluster stability.

If an ECS is deleted, the ECS OS is modified or reinstalled, or the ECS specifications are modified on the ECS console, MRS will automatically identify and delete the node. You can log in to the MRS console and restore the deleted node through scale-out. Do not perform operations on the nodes that are being scaled out.

15.13.10 How Do I Shield Cluster Alarm/Event Notifications?

1. Log in to the MRS console.
2. Click the name of the cluster.
3. On the page displayed, choose **Alarms** > **Notification Rules**.
4. Locate the row that contains the rule you want to modify, click **Edit** in the **Operation** column, and deselect the alarm or event severity levels.
5. Click **OK**.

15.13.11 Why Is the Resource Pool Memory Displayed in the MRS Cluster Smaller Than the Actual Cluster Memory?

In an MRS cluster, MRS allocates 50% of the cluster memory to Yarn by default. You manage Yarn nodes logically by resource pool. Therefore, the total memory of the resource pool displayed in the cluster is only 50% of the total memory of the cluster.

15.13.12 How Do I Configure the knox Memory?

Step 1 Log in to a Master node of the cluster as user **root**.

Step 2 Run the following command on the Master node to open the **gateway.sh** file:

su omm

vim /opt/knox/bin/gateway.sh

Step 3 Change **APP_MEM_OPTS=""** to **APP_MEM_OPTS="-Xms256m -Xmx768m"**, save the file, and exit.

Step 4 Run the following command on the Master node to restart the knox process:

sh /opt/knox/bin/gateway.sh stop

sh /opt/knox/bin/gateway.sh start

Step 5 Repeat the preceding steps on each Master node.

Step 6 Run the **ps -ef |grep knox** command to check the configured memory.

Figure 15-8 knox memory

----End

15.13.13 What Is the Python Version Installed for an MRS Cluster?

Log in to a Master node as user **root** and run the **Python3** command to query the Python version.

15.13.14 How Do I View the Configuration File Directory of Each Component?

The configuration file paths of commonly used components are as follows:

Component	Configuration File Directory
ClickHouse	*Client installation directory*/**ClickHouse/clickhouse/config**
Flink	*Client installation directory*/**Flink/flink/conf**
Flume	*Client installation directory*/**fusioninsight-flume-***xxx*/**conf**
HBase	*Client installation directory*/**HBase/hbase/conf**
HDFS	*Client installation directory*/**HDFS/hadoop/logs/hadoop.log**
Hive	*Client installation directory*/**Hive/config**
Hudi	*Client installation directory*/**Hudi/hudi/conf**
Kafka	*Client installation directory*/**Kafka/kafka/config**
Loader	• *Client installation directory*/**Loader/loader-tools-xxx/loader-tool/conf** • *Client installation directory*/**Loader/loader-tools-xxx/schedule-tool/conf** • *Client installation directory*/**Loader/loader-tools-xxx/shell-client/conf** • *Client installation directory*/**Loader/loader-tools-xxx/sqoop-shell/conf**
Oozie	*Client installation directory*/**Oozie/oozie-client-***xxx*/**conf**

Component	Configuration File Directory
Spark2x	*Client installation directory/Spark2x/spark/conf*
Yarn	*Client installation directory/Yarn/config*
ZooKeeper	*Client installation directory/Zookeeper/zookeeper/conf*

15.13.15 How Do I Do If the Time on MRS Nodes Is Incorrect?

- If the time on a node inside the cluster is incorrect, log in to the node and rectify the fault from 2.
- If the time on a node inside the cluster is different from that on a node outside the cluster, log in to the node and rectify the fault from 1.

1. Run the **vi /etc/ntp.conf** command to edit the NTP client configuration file, add the IP addresses of the master node in the MRS cluster, and comment out the IP address of other servers.

   ```
   server master1_ip prefer
   server master2_ip
   ```

   Figure 15-9 Adding the master node IP addresses

   ```
   # For more information about this file, see the man pages
   # ntp.conf(5), ntp_acc(5), ntp_auth(5), ntp_clock(5), ntp_misc(5), ntp_mon(5).

   driftfile /var/lib/ntp/drift

   # Permit time synchronization with our time source, but do not
   # permit the source to query or modify the service on this system.
   restrict default nomodify notrap nopeer noquery

   # Permit all access over the loopback interface.  This could
   # be tightened as well, but to do so would effect some of
   # the administrative functions.
   restrict 127.0.0.1
   restrict ::1

   # Hosts on local network are less restricted.
   #restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap

   # Use public servers from the pool.ntp.org project.
   # Please consider joining the pool (http://www.pool.ntp.org/join.html).
   #server 0.centos.pool.ntp.org iburst
   #server 1.centos.pool.ntp.org iburst
   #server 2.centos.pool.ntp.org iburst
   #server 3.centos.pool.ntp.org iburst
   #
   server 10.9.2.38 prefer
   server 10.9.2.39
   #broadcast 192.168.1.255 autokey        # broadcast server
   #broadcastclient                        # broadcast client
   #broadcast 224.0.1.1 autokey            # multicast server
   #multicastclient 224.0.1.1              # multicast client
   #manycastserver 239.255.254.254         # manycast server
   #manycastclient 239.255.254.254 autokey # manycast client

   # Enable public key cryptography.
   #crypto
   ```

2. Run the **service ntpd stop** command to stop the NTP service.
3. Run the **/usr/sbin/ntpdate** *IP address of the active master node* command to manually synchronize time.

4. Run the **service ntpd start** or **systemctl restart ntpd** command to start the NTP service.
5. Run the **ntpstat** command to check the time synchronization result:

15.13.16 How Do I Query the Startup Time of an MRS Node?

Log in to the target node and run the following command to query the startup time:

date -d "$(awk -F. '{print $1}' /proc/uptime) second ago" +"%Y-%m-%d %H:%M:%S"

```
[root@server-2110082001-0018 ~]#date -d "$(awk -F. '{print $1}' /proc/uptime) second ago" +"%Y-%m-%d %H:%M:%S"
2021-12-13 15:56:23
```

15.13.17 How Do I Do If Trust Relationships Between Nodes Are Abnormal?

If "ALM-12066 Inter-Node Mutual Trust Fails" is reported on Manager or there is no SSH trust relationship between nodes, rectify the fault by performing the following operations:

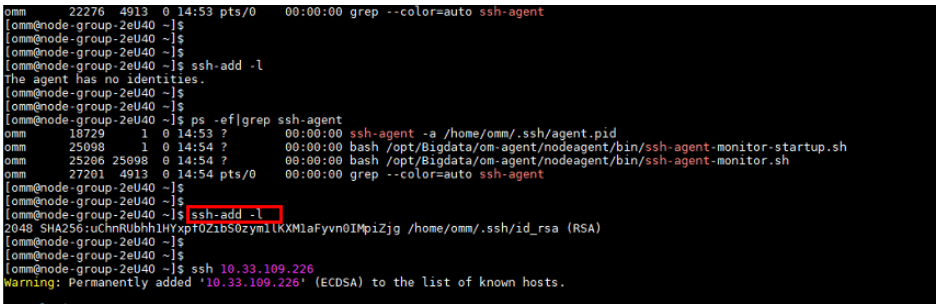
1. Run the **ssh-add -l** command on both nodes of the trusted cluster to check whether there are identities.

```
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$ ll .ssh/
total 32
srw-------. 1 omm wheel     0 Dec 29 14:17 agent.pid
-rw-------. 1 omm wheel 12901 Mar  9 14:48 authorized_keys
-rw-------. 1 omm wheel    54 Sep 24 11:42 config
-rw-------. 1 omm wheel  1766 Sep 24 11:43 id_rsa
-rw-------. 1 omm wheel   402 Sep 24 11:42 id_rsa.pub
-rw-------. 1 omm wheel    88 Jun  8  2020 id_rsa.sha256
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$ ssh-add -l
The agent has no identities.
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$ vim /var/log/Bigdata/nodeagent/
agentlog/   alarmlog/   monitorlog/ scriptlog/
[omm@node-group-2eU40 ~]$ vim /var/log/Bigdata/nodeagent/scriptlog/
agent_alarm_py.log                      install.log
agent_alarm_py.log.1                    installntp.log
```

2. If no identities are displayed, run the **ps -ef|grep ssh-agent** command to find the ssh-agent process, kill the process, and wait for the process to automatically restart.

```
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$ ssh-add -l
The agent has no identities.
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$ ps -ef|grep ssh-agent
omm      18729     1  0 14:53 ?        00:00:00 ssh-agent -a /home/omm/.ssh/agent.pid
omm      25098     1  0 14:54 ?        00:00:00 bash /opt/Bigdata/om-agent/nodeagent/bin/ssh-agent-monitor-startup.sh
omm      25206 25098  0 14:54 ?        00:00:00 bash /opt/Bigdata/om-agent/nodeagent/bin/ssh-agent-monitor.sh
omm      27201  4913  0 14:54 pts/0    00:00:00 grep --color=auto ssh-agent
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$
[omm@node-group-2eU40 ~]$ ssh-add -l
```

3. Run the **ssh-add -l** command to check whether the identities have been added. If yes, manually run the **ssh** command to check whether the trust relationship is normal.



4. If identities exist, check whether the **authorized_keys** file in the **/home/omm/.ssh** directory contains the information in the **id_rsa.pub** file in the **/home/omm/.ssh** of the peer node. If no, manually add the information about the peer node.
5. Check whether the permissions on the files in **/home/omm/.ssh** directory are correct.
6. Check the **/var/log/Bigdata/nodeagent/scriptlog/ssh-agent-monitor.log** file.
7. If the **home** directory of user **omm** is deleted, contact MRS support personnel.

15.13.18 How Do I Adjust the Memory Size of the manager-executor Process?

Symptom

The **manager-executor** process runs either on the Master1 or Master2 node in the MRS cluster in active/standby mode. This process is used to encapsulate the MRS management and control plane's operations on the MRS cluster, such as job submission, heartbeat reporting, certain alarm reporting, as well as cluster creation, scale-out, and scale-in. When you submit jobs on the MRS management and control plane, the Executor memory may become insufficient as the tasks increase or the number of concurrent tasks increases. As a result, the CPU usage is high and the Executor process experiences out-of-memory (OOM) errors.

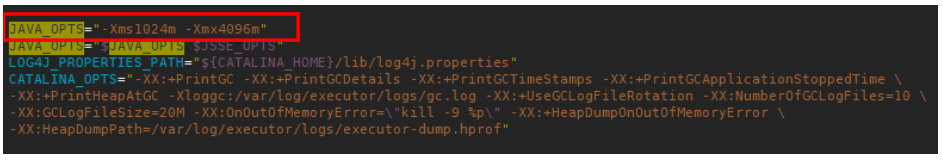
Procedure

1. Log in to either the Master1 or Master2 node as user **root** and run the following command to switch to user **omm**:

   su - omm

2. Run the following command to modify the **catalina.sh** script. Specifically, search for **JAVA_OPTS** in the script, find the configuration items similar to **JAVA_OPTS="-Xms1024m -Xmx4096m**, and change the values of the items to desired ones, and save the modification.

   vim /opt/executor/bin/catalina.sh



3. The **manager-executor** process only runs on either the Master1 or Master2 node in active/standby mode. Check whether it exists on the node before restarting it.

   a. Log in to the Master1 and Master2 nodes and run the following command to check whether the process exists. If any command output is displayed, the process exists.

      ps -ef | grep "/opt/executor" | grep -v grep

   b. Run the following command to restart the process:

      sh /opt/executor/bin/shutdown.shsh /opt/executor/bin/startup.sh

15.14 Kerberos Usage

15.14.1 How Do I Change the Kerberos Authentication Status of a Created MRS Cluster?

You cannot change the Kerberos service after an MRS cluster is created.

15.14.2 What Are the Ports of the Kerberos Authentication Service?

The Kerberos authentication service uses ports 21730 (TCP), 21731 (TCP/UDP), and 21732 (TCP/UDP).

15.14.3 How Do I Deploy the Kerberos Service in a Running Cluster?

The MRS cluster does not support customized Kerberos installation and deployment, and the Kerberos authentication cannot be set up between components. To enable Kerberos authentication, you need to create a cluster with Kerberos enabled and migrate data.

15.14.4 How Do I Access Hive in a Cluster with Kerberos Authentication Enabled?

1. Log in to the master node in the cluster as user **root**.
2. Run the following command to configure environment variables:

   source /opt/client/bigdata_env

3. If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user:

   kinit *MRS cluster user*

   Example: **kinit hiveuser**

   The current user must have the permission to create Hive tables..

4. Run the client command of the Hive component.

beeline

5. Run the Hive command in Beeline, for example:

   create table test_obs(a int, b string) row format delimited fields terminated by "," stored as textfile location "obs://test_obs";

6. Press **Ctrl+C** to exit the Hive Beeline.

15.14.5 How Do I Access Presto in a Cluster with Kerberos Authentication Enabled?

1. Log in to the Master node in the cluster as user **root**.
2. Run the following command to configure environment variables:

   source /opt/client/bigdata_env

3. Access Presto in a cluster with Kerberos authentication enabled.
   a. Log in to MRS Manager and create a role with the **Hive Admin Privilege** permission, for example, **prestorerole**.
   b. Create a user, for example, **presto001**, who belongs to the **Presto** and **Hive** groups, and bind the user to the role created in **3.a**.
   c. Authenticate user **presto001**.

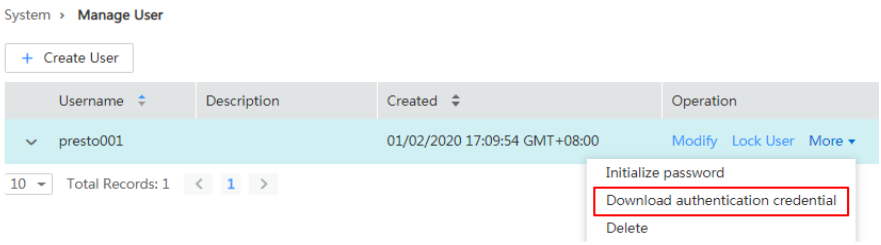
      kinit presto001

   d. Download the user authentication credential.

      - Operations on MRS Manager:

        Log in to MRS Manager, choose **System** > **Manage User**. Locate the user, and choose **More** > **Download authentication credential**.

        Figure 15-10 Downloading the Presto user authentication credential

      - Operations on FusionInsight Manager:

        Log in to FusionInsight Manager, choose **System** > **Permission** > **User**. On the displayed page, locate the row that contains the user, choose **More** > **Download Authentication Credential**.

   e. Decompress the downloaded user credential file, and save the obtained **krb5.conf** and **user.keytab** files to the client directory, for example, /opt/**client/Presto/**.
   f. Run the following command to obtain the user principal:

      klist -kt /opt/client/Presto/user.keytab

   g. Run the following command to connect to the Presto Server of the cluster:

presto_cli.sh --krb5-config-path *{krb5.conf file path}* **--krb5-principal** *{User's principal}* **--krb5-keytab-path** *{user.keytab file path}* **--user** *{presto username}*

- **krb5.conf** *file path*: file path set in **3.e**, for example, **/opt/client/Presto/krb5.conf**.
- **user.keytab** *file path*: file path set in **3.e**, for example, **/opt/client/Presto/user.keytab**.
- *User's principal*: principal obtained in **3.f**.
- *presto username*: user created in **3.b**, for example, **presto001**.

Example: **presto_cli.sh --krb5-config-path /opt/client/Presto/krb5.conf --krb5-principal prest001@xxx_xxx_xxx_xxx.COM --krb5-keytab-path /opt/client/Presto/user.keytab --user presto001**

h. On the Presto client, run the following statement to create a schema:

CREATE SCHEMA hive.demo01 WITH (location = 'obs://presto-demo002/');

i. Create a table in the schema. The table data is stored in the OBS bucket, as shown in the following example:

CREATE TABLE hive.demo01.demo_table WITH (format = 'ORC') AS SELECT * FROM tpch.sf1.customer;

Figure 15-11 Return result

j. Run **exit** to exit the client.

15.14.6 How Do I Access Spark in a Cluster with Kerberos Authentication Enabled?

1. Log in to the master node in the cluster as user **root**.
2. Run the following command to configure environment variables:

   source /opt/client/bigdata_env
3. If the Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user.

   kinit *MRS cluster user*

   Example:

   If the development user is a machine-machine user, run **kinit -kt user.keytab sparkuser**.

   If the development user is a human-machine user, run **kinit sparkuser**.
4. Run the following command to connect to Spark Beeline:

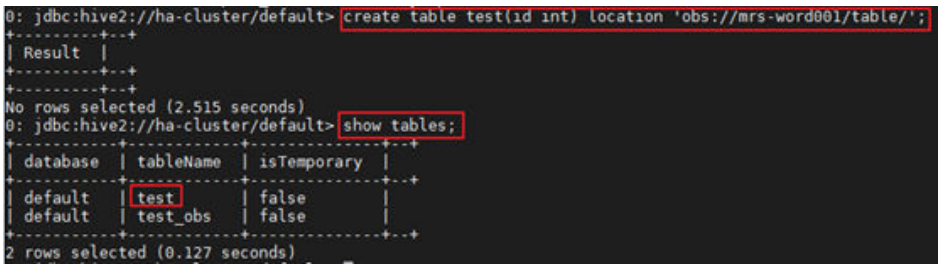
   spark-beeline
5. Run commands on Spark Beeline. For example, create the table **test** in the **obs://mrs-word001/table/** directory.

   create table test(id int) location 'obs://mrs-word001/table/';

6. Run the following command to query all tables. If table **test** is displayed in the command output, OBS access is successful.

   show tables;

   Figure 15-12 Returned table name



7. Press **Ctrl+C** to exit Spark Beeline.

15.14.7 How Do I Prevent Kerberos Authentication Expiration?

- Java applications:

  Before connecting to HBase, HDFS, or other big data components, call loginUserFromKeytab() to create a UGI. Then, start a scheduled thread to periodically check whether the Kerberos Authentication expires. Log in to the system again before the Kerberos Authentication expires.

```
private static void startCheckKeytabTgtAndReloginJob() {
//The credential is checked every 10 minutes, and updated before the expiration time.
      ThreadPool.updateConfigThread.scheduleWithFixedDelay(() -> {
         try {
            UserGroupInformation.getLoginUser().checkTGTAndReloginFromKeytab();
            logger.warn("get tgt:{}", UserGroupInformation.getLoginUser().getTGT());
            logger.warn("Check Kerberos Tgt And Relogin From Keytab Finish.");
         } catch (IOException e) {
            logger.error("Check Kerberos Tgt And Relogin From Keytab Error", e);
         }
      }, 0, 10, TimeUnit.MINUTES);
      logger.warn("Start Check Keytab TGT And Relogin Job Success.");
   }
```

- Tasks executed in shell mode:

  a. Run the **kinit** command to authenticate the user.

  b. Create a scheduled task of the operating system or any other scheduled task to run the **kinit** command to authenticate the user periodically.

  c. Submit jobs to execute big data tasks.

- Spark jobs:

  If you submit jobs using spark-shell, spark-submit, or spark-sql, you can specify **Keytab** and **Principal** in the command to perform authentication and periodically update the login credential and authorization tokens to prevent authentication expiration.

  Example:

  spark-shell --principal spark2x/hadoop.<*System domain name*>@<*System domain name*> **--keytab ${BIGDATA_HOME}/FusionInsight_Spark2x_8.1.0.1/install/FusionInsight-Spark2x-2.4.5/keytab/spark2x/SparkResource/spark2x.keytab --master yarn**

15.15 Metadata Management

15.15.1 Where Can I View Hive Metadata?

- If Hive metadata is stored in GaussDB of an MRS cluster, log in to the master DBServer node of the cluster, switch to user **omm**, and run the **gsql -p 20051 -U {USER} -W {PASSWD} -d hivemeta** command to view the metadata.
- If Hive metadata is stored in an external relational database, perform the following steps:
  a. On the cluster **Dashboard** page, click **Manage** on the right of **Data Connection**.
  b. On the displayed page, obtain the value of **Data Connection ID**.
  c. On the MRS console, click **Data Connections**.
  d. In the data connection list, locate the data connection based on the data connection ID obtained in **b**.
  e. Click **Edit** in the **Operation** column of the data connection.

     The **RDS Instance** and **Database** indicate the relational database in which the Hive metadata is stored.

16 Troubleshooting

16.1 Accessing the Web Pages

16.1.1 Failed to Access MRS Manager

Symptom

The MRS Manager is not accessible after a cluster is created.

Possible Cause

- MRS can be accessed from an external network only after an EIP is bound to an MRS node.
- Port 9022 is disabled. Add a security group rule to enable the port.

Procedure

Step 1 Log in to the MRS management console, locate the cluster to be accessed in the active cluster list, and click the cluster name.

Step 2 On the node information page, click the name of the node to be accessed, and choose **EIPs** > **Bind EIP**.

Step 3 On the **Bind EIP** page, select a NIC from the **Select NIC** drop-down list, select an EIP from the **Select EIP** list, and click **OK**.

Step 4 After the EIP is bound, enable port 9022 in a security group rule.

Click the **Security Groups** tab. Then, click **Change Security Group**.

You can select an existing security group, or click **Create Security Group** to add a security group rule to enable port 9022 for accessing through the public IP address.

Step 5 After the EIP is added, you can access MRS through **https://***Elastic IP address*:**9022/mrsmanager/**. If the fault still persists, contact technical support for assistance.

----End

16.1.2 Failed to Log In to MRS Manager After the Python Upgrade

Issue

Failed to log in to MRS Manager after Python is upgraded.

Symptom

After Python is upgraded, MRS Manager fails to be accessed using the **admin** account and the correct password.

Possible Cause

When upgrading Python to Python 3.*x*, the user modifies the file directory permission of **openssl**. As a result, the LdapServer service cannot be started, causing a login authentication failure.

Procedure

Step 1 Log in to the Master node in the cluster as user **root**.

Step 2 Run the **chmod 755 /usr/bin/openssl** command to modify the file directory permission of **/usr/bin/openssl** to **755**.

Step 3 Run the **su omm** command to switch to user **omm**.

Step 4 Run the **openssl** command to check whether the **openssl** mode can be entered.

If it can be entered, the permission has been modified successfully. If it cannot be entered, the permission fails to be modified.

If the permission fails to be modified, check whether the command is correct or contact O&M personnel.

Step 5 After the permission is modified, the LdapServer service will be restarted. After the LdapServer service is restarted, log in to MRS Manager again.

----End

Summary and Suggestions

It is recommended that software installed by the user be separated from system software. A system software upgrade may cause compatibility problems.

16.1.3 Failed to Log In to MRS Manager After Changing the Domain Name

Symptom

After changing the domain name, the user cannot log in to MRS Manager through the console, or fails to log in to MRS Manager.

Possible Causes

After the domain name is changed, the **keytab** file of user **executor** is not updated. As a result, the executor process repeatedly performs authentication after the authentication fails, causing memory overflow of the ACS process.

Solution

Step 1 Restart the acs process.

1. Log in to the active management node (master node marked a solid star on the **Nodes** tab of the MRS cluster) as user **root**.
2. Run the following commands to restart the acs process:

   su - omm

   ps -ef|grep =acs (Query the PID of the acs process.)

   kill -9 *PID* (Replace *PID* with the acs process ID to kill the acs process.)
3. Wait for several minutes and run the **ps -ef|grep =acs** command to check whether the acs process is automatically started.

Step 2 Replace the **keytab** file of user **executor**.

1. Log in to MRS Manager and choose **System** > **User**. In the **Operation** column where user **executor** resides, click **Download Authentication Credential**. Decompress the package to obtain the **keytab** file.
2. Log in to the active management node as user **root** and replace the **/opt/executor/webapps/executor/WEB-INF/classes/user.keytab** file with the file obtained in **Step 2.1**.

Step 3 Replace the **keytab** and **conf** files of user **knox**.

1. Log in to MRS Manager and choose **System** > **User**. In the **Operation** column where user **knox** resides, click **Download Authentication Credential**. Decompress the package to obtain the **keytab** and **conf** files.
2. Log in to the active management node as user **root** and replace the **/opt/knox/conf/user.keytab** with the file obtained in **Step 3.1**.
3. Change the **principal** value in the **/opt/knox/conf/krb5JAASLogin.conf** file to the new domain name.
4. Replace the **/opt/knox/conf/krb5.conf** file with the **krb5.conf** file obtained in **Step 3.1**.

Step 4 Back up the original client directory.

mv *{Client directory}* **/opt/client_init**

Step 5 Reinstall the client.

Step 6 Log in to the active and standby management nodes as user **root** and run the following commands to restart the knox process:

su - omm

ps -ef | grep gateway | grep -v grep (Search for the PID of the knox process.)

kill -9 *PID* (Replace *PID* with the ID of the knox process to kill the knox process.)

/opt/knox/bin/restart-knox.sh (Start the knox process.)

Step 7 Log in to the active and standby management nodes as user **root** and run the following commands to restart the executor process:

su - omm

netstat -anp |grep 8181 |grep LISTEN (Search for the PID of the executor process.)

kill -9 *PID* (Replace *PID* with the ID of the executor process to kill the executor process.)

/opt/executor/bin/startup.sh (Start the executor process.)

----End

16.1.4 A Blank Page Is Displayed Upon Login to Manager

Issue

After a user logs in to FusionInsight Manager, the page displayed is blank.

Symptom

After a user logs in to FusionInsight Manager, the page displayed is blank.

Cause Analysis

Login to FusionInsight Manager fails, and the browser cache needs to be cleared.

Procedure

Step 1 Open the browser (using Google Chrome as an example), and press **Ctrl+Shift +Delete**. The dialog box for clearing browsing data is displayed.

Step 2 Select the browsing records to be cleared and click **Clear Data**.

----End

16.1.5 Failed to Download Authentication Credentials When the Username Is Too Long

Issue

In MRS clusters 3.0.2 to 3.1.0, a maximum of 32 characters are allowed in the username when a user is added. However, if the username contains more than 20

characters, the user fails to download the Keytab file, and status code "400 Bad Request" is displayed.

Symptom

In MRS clusters 3.0.2 to 3.1.0, a maximum of 32 characters are allowed in the username when a user is added. However, if the username contains more than 20 characters, the user fails to download the Keytab file, and status code "400 Bad Request" is displayed.

Cause Analysis

The **validate-common-config.xml**, **validate-rule-session.xml**, and **validate-rule-user.xml** configuration files in the **/opt/Bigdata/om-server_*/apache-tomcat-*/webapps/web/WEB-INF/validate** directory of the master node are incorrect and need to be modified.

Procedure

Step 1 Log in to the master node as user **omm** and switch to the **/opt/Bigdata/om-server_*/apache-tomcat-*/webapps/web/WEB-INF/validate** directory.

cd /opt/Bigdata/om-server_*/apache-tomcat-*/webapps/web/WEB-INF/validate

Step 2 Modify the **validate-common-config.xml** file.

vi validate-common-config.xml

Change the **maxLength** value of the username from **32** to **64**.

```
<!-- Username -->
<validators alias="USER_NAME">
      <validator name="RANGE_LENGTH_VALIDATOR" minLength="3"
         maxLength="64" />
      <validator name="REGEXP_VALIDATOR" rule="^[_a-zA-Z0-9\- ]+$"
</validators>
```

Step 3 Modify the **validate-rule-session.xml** file.

vi validate-rule-session.xml

Change the **rule** value from **20** to **64**.

```
 <!-- Download the credentials of the current user -->
 <param_validator url="/api/v2/session/user/keytab/download" method="get"
errorHandler="com.xxx.bigdata.om.web.api.validate.SpecialValidatorErrorHandler" dataPattern="form">
            <!-- Parameter name: File name -->
            <!--Validation rule: userName_13-digit number_keytab.tar; case sensitive-->
            <parameter name="file_name" required="true" errorKey="13-4000005"
errorMessage="RESID_OM_API_SESSION_0013">
                  <validator name="REGEXP_VALIDATOR" rule="[\-\w ]{3,64}_\d{13}_keytab\.tar"
caseSensitive="true" />
 </parameter>
```

Step 4 Modify the **validate-rule-user.xml** file.

vi validate-rule-user.xml

Change the **rule** value from **20** to **64**.

```
<!--Download the user credentials  -->
  <param_validator url="/api/v2/permission/users/keytab/download" method="get"
```

```
errorHandler="com.xxx.bigdata.om.web.api.validate.SpecialValidatorErrorHandler" dataPattern="form">
            <!--Mandatory; userName_13-digit number_keytab.tar; case sensitive-->
            <parameter name="file_name" required="true" errorKey="12-4000005"
errorMessage="RESID_OM_API_AUTHORITY_0005">
                    <validator name="REGEXP_VALIDATOR" rule="[\-\w ]{3,64}_\d{13}_keytab\.tar"
caseSensitive="true" />
            </parameter>
    </param_validator>
```

Step 5 Restart Tomcat and wait until the startup is successful.

1. Run the following command as user **omm** to query the PID of the Tomcat process:

   ps -ef|grep apache-tomcat

2. Run the **kill -9** *PID* command to forcibly stop the specified Tomcat process. For example:

   kill -9 1203

3. Run the following command to restart Tomcat:

   sh ${BIGDATA_HOME}/om-server/tomcat/bin/startup.sh

Step 6 Download the authentication credentials again.

----End

16.2 Cluster Management

16.2.1 Failed to Reduce Task Nodes

Issue

A user fails to scale in an MRS 2.*x* cluster by reducing the number of task nodes to **0** on the MRS console.

Symptom

When the number of task nodes in an MRS cluster is reduced on the MRS console, the following information is displayed:

This operation is not allowed because the number of instances of NodeManager will be less than the minimum configuration after scale-in, which may cause data loss.

Cause Analysis

The NodeManager service of the core node is stopped. If the number of task nodes is changed to **0**, there will be no NodeManager in the cluster and the Yarn service will be unavailable. Therefore, MRS allows the reduction of task nodes only when the number of NodeManagers is greater than or equal to **1**.

Procedure

Step 1 Select the NodeManager instance of the core node, click **More**, and select **Start Instance**.

Step 2 Reduce the number of task nodes on the cluster details page.

1. Click the cluster name, and select the **Nodes** tab.
2. Locate the row that contains the task node group and click **Scale In** in the **Operation** column.
3. Click **OK**. In the displayed dialog box, click **Yes**.

Step 3 After the scale-in is successful, stop NodeManager of the core node if you do not need it.

----End

Summary and Suggestions

You are advised not to stop NodeManager of the core node.

16.2.2 Adding a New Disk to an MRS Cluster

Issue

MRS HBase is unavailable.

Symptom

A high disk usage of the user's host causes service faults.

Cause Analysis

The service becomes unavailable due to insufficient disk capacity of the core node.

Procedure

Step 1 Purchase an EVS disk.

Step 2 Attach the EVS disk.

- If the EVS disk has been attached, go to **Step 6**.
- If an ECS cannot be selected when you attach the EVS disk on the EVS console, go to **Step 3**.

Step 3 Log in to the ECS console and click the name of the ECS to which the new disk is to be attached.

Step 4 On the **Disks** tab, click **Attach Disk**.

Step 5 Select the new disk to be attached and click **OK**.

Step 6 Initialize a Linux data disk.

NOTE

- The mount point directory is the existing DataNode instance ID plus one. For example, if you run the **df –h** command and find that the existing ID is **/srv/BigData/hadoop/data1**, the new mount point is then **/srv/BigData/hadoop/data2**. When initializing a Linux data disk to create a mount point, name the mount point **/srv/BigData/hadoop/data2** and mount a new partition to the mount point. For example:
  ```
  mkdir /srv/BigData/hadoop/data2
  mount /dev/xvdb1 /srv/BigData/hadoop/data2
  ```

About the **/srv/BigData/hadoop/data2** path: Change **/srv/BigData/hadoop/data2** mentioned below according to the following scenarios:

- In 3.*x*: Change it to **/srv/BigData/data2**.
- In versions earlier than 3.*x* : Change it to **/srv/BigData/hadoop/data2**.

Step 7 Run the following command to grant user **omm** the permissions to access the new disk:

chown omm:wheel *New mount point*

Example: **chown omm:wheel /srv/BigData/hadoop/data2**

Step 8 Run the following command to grant the execution permission on the new mount point directory:

chmod 701 *New mount point*

Example: **chmod 701 /srv/BigData/hadoop/data2**

NOTE

In this command, **701** is only an example. Replace it with the value of the existing data disk **data1**.

Step 9 Log in to Manager and add data disks to DataNode and NodeManager instances.

Step 10 Modify the DataNode instance configuration.

MRS Manager: Log in to MRS Manager, choose **Services** > **HDFS** > **Instance**, click the target DataNode instance, and click **Instance Configuration**. On the displayed page, set **Type** to **All**.

FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster**. Click the name of the desired cluster and choose **Service** > **HDFS** > **Instance**. Click the target DataNode instance, click **Instance Configuration**, and select **All Configurations**.

- Method 1: Manually modify the DataNode instance configuration on the current node.
  - Enter **dfs.datanode.fsdataset.volume.choosing.policy** in the search box and change the parameter value to **org.apache.hadoop.hdfs.server.datanode.fsdataset.AvailableSpaceVolumeChoosingPolicy**.
  - Enter **dfs.datanode.data.dir** in the search box and change the parameter value to **/srv/BigData/hadoop/data1/dn,/srv/BigData/hadoop/data2/dn**.

  If the values of the two parameters have been changed, click **Save Configuration** and select **Restart role instance** to restart the DataNode instance.

- Method 2: Automatically synchronize the DataNode instance configuration on the current node.
  a. Click **Synchronize Configuration** to enable the new configuration for the HDFS service.
  b. After the synchronization is complete, restart the instance for the configuration to take effect.

NOTE

- If HDFS is not used and you want to quickly restart the instance, select **Restart role instance**.
- If a task is using HDFS, you must select rolling restart to prevent data exceptions or task failures.

Step 11 Modify the Yarn NodeManager instance configuration.

MRS Manager: Log in to MRS Manager, choose **Services** > **Yarn** > **Instance**, click the target NodeManager instance, and click **Instance Configuration**. On the displayed page, set **Type** to **All**.

FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster**. Click the name of the desired cluster and choose **Service** > **Yarn** > **Instance**. Click the target NodeManager instance, click **Instance Configuration**, and select **All Configurations**.

- Method 1: Manually modify the Yarn NodeManager instance configuration on the current node.
  - Enter **yarn.nodemanager.local-dirs** in the search box and change the parameter value to **/srv/BigData/hadoop/data1/nm/localdir,/srv/BigData/hadoop/data2/nm/localdir**.
  - Enter **yarn.nodemanager.log-dirs** in the search box and change the parameter value to **/srv/BigData/hadoop/data1/nm/containerlogs,/srv/BigData/hadoop/data2/nm/containerlogs**.

    If the values of the two parameters have been changed, click **Save Configuration** and select **Restart role instance** to restart the NodeManager instance.
- Method 2: Automatically synchronize the Yarn NodeManager instance configuration on the current node.
  a. Click **Synchronize Configuration** to enable the new configuration for the Yarn service.
  b. After the synchronization is complete, restart the instance for the configuration to take effect.

NOTE

- If Yarn is not used and you want to quickly restart the instance, select **Restart role instance**.
- If a task is using Yarn, you must select rolling restart to prevent data exceptions or task failures.

Step 12 Check whether the capacity expansion is successful.

MRS Manager: Log in to MRS Manager, choose **Services** > **HDFS** > **Instance**, and click the target DataNode instance.

FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster**. Click the name of the desired cluster, choose **Service** > **HDFS** > **Instance**, and click the target DataNode instance.

In the **Chart** area, check whether the total disk capacity in real-time monitoring item **DataNode Storage** is increased. If **DataNode Storage** does not exist in the **Chart** area, click **Customize** to add it.

- If the total disk capacity has been increased, the capacity expansion is complete.
- If the total disk capacity does not increase, contact technical support.

Step 13 (Optional) Add data disks to a Kafka instance.

Modify the Kafka instance configuration.

1. Navigate to the parameter settings of the target Kafka Broker node.

   MRS Manager: Log in to MRS Manager, choose **Services** > **Kafka** > **Instance**, click the target Broker instance, and click **Instance Configuration**. On the displayed page, set **Type** to **All**.

   FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster**. Click the name of the desired cluster and choose **Service** > **Kafka** > **Instance**. Click the target Broker instance, click **Instance Configuration**, and select **All Configurations**.
2. Enter **log.dirs** in the search box, add information about the disks to be added, and use commas (,) to separate them.

   For example, if there is only one existing Kafka data disk and a new one is added, change **/srv/BigData/kafka/data1/kafka-logs** to **/srv/BigData/kafka/data1/kafka-logs,/srv/BigData/kafka/data2/kafka-logs**.
3. Save the configuration and select **Restart role instance** to restart the instance as prompted.
4. Check whether the capacity expansion is successful.

   MRS Manager: Log in to MRS Manager, choose **Services** > **Kafka** > **Instance**, and click the target Broker instance.

   FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster**. Click the name of the desired cluster, choose **Service** > **Kafka** > **Instance**, and click the target Broker instance.

   Check whether the total disk capacity in real-time monitoring item **Capacity of Broker Disks** is increased.

----End

NOTICE

After the disk capacity of a cluster node is expanded, if a new node is added to the cluster, you need to add disks to the new node by referring to the preceding procedure. Otherwise, data may be lost.

Summary and Suggestions

- If the disk usage exceeds 85%, you are advised to expand disk capacity and attach the newly purchased disks to ECSs to associate with the cluster.

- The procedure for attaching disks and setting parameters may vary depending on the site environment.

16.2.3 Replacing a Disk in an MRS Cluster (Applicable to 2.x and Earlier)

Issue

A disk is not accessible.

Symptom

A user created an MRS cluster with local disks. A disk of a core node in this cluster is damaged, resulting in file read failures.

Cause Analysis

The disk hardware is faulty.

Procedure

NOTE

This procedure is applicable to analysis clusters earlier than MRS 3.*x*. If you need to replace disks for a streaming cluster or hybrid cluster, contact technical support.

Step 1 Log in to .

Step 2 Choose **Hosts**, click the name of the host to be decommissioned, click **RegionServer** in the **Roles** list, click **More**, and select **Decommission**.

Step 3 Choose **Hosts**, click the name of the host to be decommissioned, click **DataNode** in the **Roles** list, click **More**, and select **Decommission**.

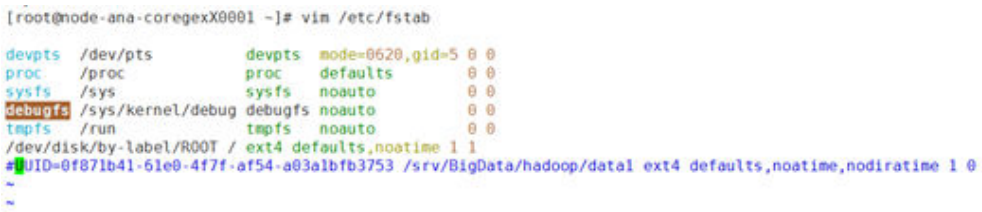
Step 4 Choose **Hosts**, click the name of the host to be decommissioned, click **NodeManager** in the **Roles** list, click **More**, and select **Decommission**.

NOTE

If this host still runs other instances, perform the similar operation to decommission the instances.

Step 5 Run the **vim /etc/fstab** command to comment out the mount point of the faulty disk.

Figure 16-1 Commenting out the mount point of the faulty disk



Step 6 Migrate the user data on the faulty disk (for example, **/srv/BigData/hadoop/data1/**).

Step 7 Log in to the MRS console.

Step 8 On the cluster details page, click the **Nodes** tab.

Step 9 Click the node whose disk is to be replaced to go to the ECS console. Click **Stop** to stop the node.

Step 10 Contact technical support to replace the disk in the background.

Step 11 On the ECS console, click **Start** to start the node where the disk has been replaced.

Step 12 Run the **fdisk -l** command to view the new disk.

Step 13 Run the **cat /etc/fstab** command to obtain the drive letter.

Figure 16-2 Obtaining the drive letter

```
[omm@node-master1dGom ~]$ cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Wed Feb 27 06:58:49 2019
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=b13ee9c8-0ef0-4159-9b90-fc47bde0d464 /                       ext4    defaults,noatime        1 1
UUID=029408e0-71a6-4f73-b817-42d7049b7595 /srv/BigData ext4 defaults,noatime,nodiratime 1 0
UUID=f9cb8844-dabf-4a69-aff4-587de2fc4d7c /srv/BigData1 ext4 defaults,noatime,nodiratime 1 0
UUID=876e73be-1f80-4466-92b7-01d7c68bbb1b /srv/BigData2 ext4 defaults,noatime,nodiratime 1 0
UUID=0d5fce7f-afd0-420a-b1bb-e5500a1851cd /srv/BigData3 ext4 defaults,noatime,nodiratime 1 0
```

Step 14 Use the corresponding drive letter to format the new disk.

Example: **mkfs.ext4 /dev/sdh**

Step 15 Run the following command to attach the new disk.

mount *New disk Mount point*

Example: **mount /dev/sdh /srv/BigData/hadoop/data1**

Step 16 Run the following command to grant the **omm** user permission to the new disk:

chown omm:wheel *Mount point*

Example: **chown -R omm:wheel /srv/BigData/hadoop/data1**

Step 17 Add the UUID of the new disk to the **fstab** file.

1. Run the **blkid** command to check the UUID of the new disk.

   ```
   [root@node-ana-coreKpoT0003 ~]# blkid
   /dev/vda1: LABEL="ROOT" UUID="2aa97872-11ec-422e-9513-0f28b925ad5e" TYPE="ext4"
   /dev/vdb: UUID="e5f652c3-f9af-427f-89da-f2545618688d" TYPE="ext4"
   [root@node-ana-coreKpoT0003 ~]#
   ```

2. Open the **/etc/fstab** file and add the following information:
   UUID=*New disk UUID* **/srv/BigData/hadoop/data1 ext4 defaults,noatime,nodiratime 1 0**

Step 18 (Optional) Create a log directory.

mkdir -p /srv/BigData/Bigdata

chown omm:ficommon /srv/BigData/Bigdata

chmod 770 /srv/BigData/Bigdata

NOTE

Run the following command to check whether symbolic links to **Bigdata** logs exist. If yes, skip this step.

ll /var/log

Step 19 Log in to .

Step 20 Choose **Hosts**, click the name of the host to be recommissioned, click **RegionServer** in the **Roles** list, click **More**, and select **Recommission**.

Step 21 Choose **Hosts**, click the name of the host to be recommissioned, click **DataNode** in the **Roles** list, click **More**, and select **Recommission**.

Step 22 Choose **Hosts**, click the name of the host to be recommissioned, click **NodeManager** in the **Roles** list, click **More**, and select **Recommission**.

NOTE

If this host still runs other instances, perform the similar operation to recommission the instances.

Step 23 Choose **Services** > **HDFS**. In the **HDFS Summary** area on the **Service Status** page, check whether **Missing Blocks** is **0**.

- If **Missing Blocks** is **0**, no further action is required.
- If **Missing Blocks** is not **0**, contact technical support.

----End

16.2.4 Replacing a Disk in an MRS Cluster (Applicable to 3.x)

Issue

A disk is not accessible.

Symptom

A user created an MRS cluster with local disks. A disk of a core node in this cluster is damaged, resulting in file read failures.

Cause Analysis

The disk hardware is faulty.

Procedure

NOTE

This procedure is applicable to troubleshooting disk hardware faults of core and task nodes in MRS clusters using local disks (ECSs of D, I, IR, and KI series).

Kafka does not support disk replacement. If the node that stores Kafka data is faulty, contact technical support.

Step 1 Log in to .

Step 2 Choose **Hosts** and click the name of the faulty host. In the **Instance** area, click **DataNode**. Then on the page that is displayed, click **More** and select **Decommission**.

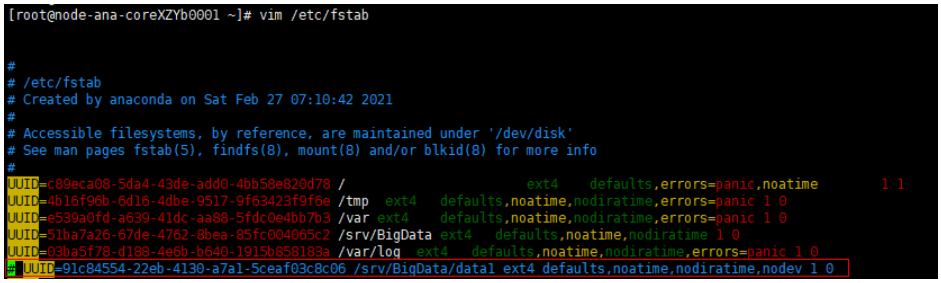
NOTE

- If this host accommodates DataNode, NodeManager, RegionServer, and ClickHouseServer instances, decommission these instances by referring to this step.
- In versions later than MRS 3.1.2, the ClickHouseServer role instance can be decommissioned.

Step 3 Choose **Hosts**, select the faulty host, click **More**, and select **Stop All Instances**.

Step 4 Run the **vim /etc/fstab** command to comment out the mount point of the faulty disk.

Figure 16-3 Commenting out the mount point of the faulty disk



Step 5 If the old disk is still accessible, migrate user data on the old disk (for example, **/srv/BigData/data1/**).

cp -r *Mount point of the old disk Temporary data storage directory*

Example: **cp -r /srv/BigData/data1 /tmp/**

Step 6 Log in to the MRS console.

Step 7 On the cluster details page, click the **Nodes** tab.

Step 8 Click the node whose disk is to be replaced to go to the ECS console. Click **Stop** to stop the node.

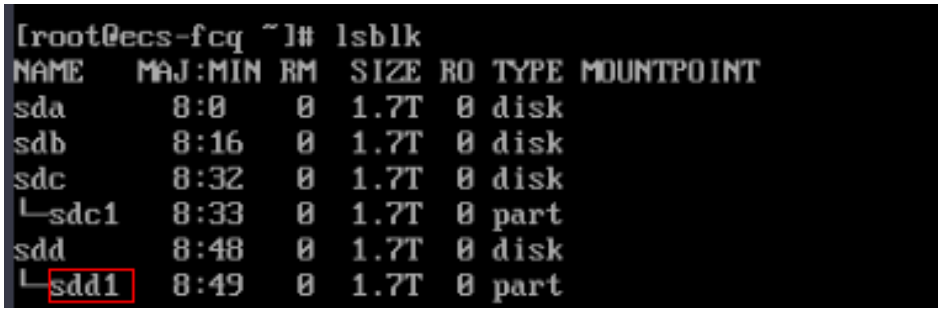
Step 9 Contact technical support to replace the disk in the background.

Step 10 On the ECS console, click **Start** to start the node where the disk has been replaced.

Step 11 Initialize the Linux data disk.

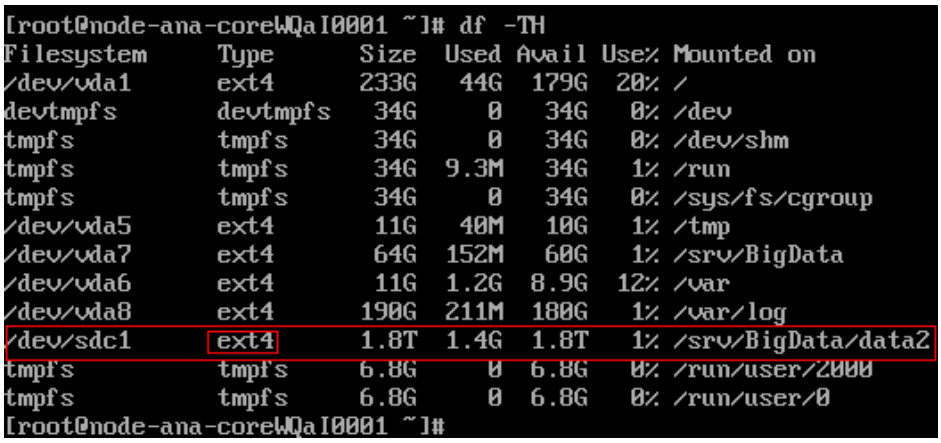
Step 12 Run the **lsblk** command to view information about the new disk partition.

Figure 16-4 Viewing the new disk partition



Step 13 Run the **df -TH** command to obtain the file system type.

Figure 16-5 Obtaining the file system type



Step 14 Format the new disk partition based on the obtained file system type.

Example: **mkfs.ext4 /dev/sdd1**

Step 15 Run the following command to mount the new disk:

mount *New disk Mount point*

Example: **mount /dev/sdd1 /srv/BigData/data1**

NOTE

If the disk cannot be mounted, run the following command to reload the configuration and mount it again:

systemctl daemon-reload

Step 16 Run the following command to grant the **omm** user permission to the new disk:

chown omm:wheel *Mount point*

Example: **chown -R omm:wheel /srv/BigData/data1**

Step 17 Migrate user data from the old disk (for example, **/srv/BigData/data1/**) to the new disk.

cp -r *Temporary data storage directory Mount point of the new disk*

Example: **cp -r /tmp/data1/* /srv/BigData/data1/**

Step 18 Add the UUID of the new disk to the **fstab** file.

1. Run the **blkid** command to check the UUID of the new disk.

2. Open the **/etc/fstab** file and add the following information:
   UUID=*UUID of the new disk* /srv/BigData/data1 ext4 defaults,noatime,nodiratime,nodev 1 0

Step 19 Log in to .

Step 20 Choose **Hosts** and click the name of the host to be recommissioned. In the **Instance** area, click **DataNode**. Then on the page that is displayed, click **More** and select **Recommission**.

NOTE

- If this host accommodates DataNode, NodeManager, RegionServer, and ClickHouseServer instances, recommission these instances by referring to this step.
- In versions later than MRS 3.1.2, the ClickHouseServer role instance can be recommissioned.

Step 21 Choose **Hosts**, select the faulty host, click **More**, and select **Start All Instances**.

Step 22 Choose **Cluster** > **HDFS**. In the **Basic Information** area on the **Dashboard** page, check whether **Missing Blocks** is **0**.

- If **Missing Blocks** is **0**, no further action is required.
- If **Missing Blocks** is not **0**, contact technical support.

----End

16.2.5 MRS Backup Failure

Issue

MRS backup keeps failing.

Symptom

MRS backup keeps failing.

Cause Analysis

The backup directory is connected to the system disk using a soft link. As a result, if the system disk is full, the backup fails.

Procedure

Step 1 Check whether the backup directory is connected to the system disk using a soft disk.

1. Log in to the active and standby Master nodes in the cluster as user **root**.
2. Run the **df -h** command to check the storage usage of the system disk.
3. Run the **ll /srv/BigData/LocalBackup** command to check whether the backup directory is connected to **/opt/Bigdata/LocalBackup** using a soft link.

   Check whether the backup file is connected to the system disk using a soft link and whether the system disk has sufficient space. If the soft link is used for connecting to the system disk and the system disk space is insufficient, go to **Step 2**. If the soft link is not used, the failure is not caused by insufficient system disk space. Contact technical support for troubleshooting.

Step 2 Move historical backup data to a new directory on the data disk.

1. Log in to the Master node as user **root**.
2. Run the **su - omm** command to switch to user **omm**.
3. Run the **rm -rf /srv/BigData/LocalBackup** command to delete the soft link of the backup directory.
4. Run the **mkdir -p /srv/BigData/LocalBackup** command to create a backup directory.
5. Run the **mv /opt/Bigdata/LocalBackup/* /srv/BigData/LocalBackup/** command to move the historical backup data to the new directory.

----End

16.2.6 Inconsistency Between df and du Command Output on the Core Node

Issue

The capacity displayed in the **df** command output on the Core node is inconsistent with that displayed in the **du** command output.

Symptom

After the **df** and **du** commands are executed, the values of the Core node capacity displayed are different.

The disk usage of the **/srv/BigData/hadoop/data1/** directory queried by running the **df -h** command differs greatly from that queried by running the **du -sh /srv/BigData/hadoop/data1/** command. The difference is greater than 10 GB.

Cause Analysis

The **lsof |grep deleted** command output indicates that a large number of log files in the directory are in the deleted state.

When some Spark tasks are running for a long time, some containers in the tasks keep running and logs are continuously generated. When printing logs, the executor of Spark uses the log4j log scrolling function to output logs to the **stdout** file. The container also monitors this file. As a result, the file is monitored by two processes at the same time. When one process scrolls according to the configuration, the earliest log file is deleted, but the other process still occupies the file handle. As a result, a file in the deleted state is generated.

Procedure

Change the output directory name for executor logs of Spark.

1. Open the log configuration file. By default, the configuration file is located in *<Client address>***/Spark/spark/conf/log4j-executor.properties**.
2. Change the name of the log output file.

   For example, change **log4j.appender.sparklog.File = ${spark.yarn.app.container.log.dir}/stdout** to **log4j.appender.sparklog.File = ${spark.yarn.app.container.log.dir}/stdout.log**.

3. Save the configuration and exit.
4. Submit the tasks again.

16.2.7 Disassociating a Subnet from the ACL Network

Scenarios

You can disassociate a subnet from the ACL network when necessary.

Procedure

Step 1 Log in to the management console.

Step 2 On the console homepage, under **Network**, click **Virtual Private Cloud**.

Step 3 In the navigation tree on the left, choose **Network ACL**.

Step 4 Locate the target network ACL in the right pane, and click the network ACL name to switch to the network ACL details page.

Step 5 On the displayed page, click the **Associated Subnets** tab.

Step 6 On the **Associated Subnets** page, locate the target network ACL and click **Disassociate** in the **Operation** column.

Step 7 Click **OK**.

----End

16.2.8 MRS Becomes Abnormal After hostname Modification

Issue

What should I do if MRS becomes abnormal after **hostname** is modified?

Symptom

MRS becomes abnormal after **hostname** is modified.

Possible Cause

The **hostname** modification causes compatibility problems and faults.

Procedure

Step 1 Log in to any node in the cluster as user **root**.

Step 2 Run the **cat /etc/hosts** command on the node to check the value of **hostname** of each node and set the **newhostname** variable based on the value.

Step 3 Run the **sudo hostnamectl set-hostname ${newhostname}** command on the node where **hostname** is modified to restore the correct hostname.

NOTE

${newhostname}: new value of **hostname**

Step 4 After the modification, log in to the node where **hostname** is modified, and check whether the new hostname takes effect.

----End

16.2.9 DataNode Restarts Unexpectedly

Symptom

A DataNode is restarted unexpectedly, but no manual restart operation is performed for the DataNode.

Cause Analysis

Possible causes:

- **OOM of the Java process is killed.**

  In general, the OMM Killer is configured for Java processes to detect and kill OOM. The OOM log is printed in the out log. In this case, you can view the run log (for example, the DataNode's log path is **/var/log/Bigdata/hdfs/dn/hadoop-omm-datanode-***hostname***.log**) to check whether OutOfMemory is printed.

- **DataNode is manually killed or killed by another process.**

  Check the DataNode run log file **/var/log/Bigdata/hdfs/dn/hadoop-omm-datanode-***hostname***.log**. It is found that the health check fails after "RECEIVED SIGNAL 15" is received. In the following example, the DataNode is killed at 11:04:48 and then started at 11:06:52 two minutes later.

  ```
  2018-12-06 11:04:48,433 | ERROR | SIGTERM handler | RECEIVED SIGNAL 15: SIGTERM | LogAdapter.java:69
  2018-12-06 11:04:48,436 | INFO  | Thread-1 | SHUTDOWN_MSG:
  /************************************************************
  SHUTDOWN_MSG: Shutting down DataNode at 192-168-235-85/192.168.235.85
  ************************************************************/ | LogAdapter.java:45
  2018-12-06 11:06:52,744 | INFO  | main | STARTUP_MSG:
  ```

  According to the logs, DataNode was closed and then the health check reported the exception. After 2 minutes, NodeAgent started the DataNode process.

Procedure

Add the rule for recording the kill command in the audit log of the operating system. The process that delivers the kill command will be recorded in the audit log.

Operation impact

- Printing audit logs affects operating system performance. However, analysis result shows that the impact is less than 1%.
- Printing audit log occupies some disk space. The logs to be printed are within megabytes. By default, the aging mechanism and the mechanism for checking the remaining disk space are configured. Therefore, the disk space will not be used up.

Locating Method

Perform the following operations on nodes that may restart the DataNode process:

Step 1 Log in to the node as the **root** user and run the **service auditd status** command to check the service status.

```
Checking for service auditd  running
```

If the service is not started, run the **service auditd restart** command to restart the service. The command execution takes less than 1 second and has no impact on the system.

```
Shutting down auditd done
Starting auditd done
```

Step 2 The audit rule of the **kill** command is temporarily added to audit logs.

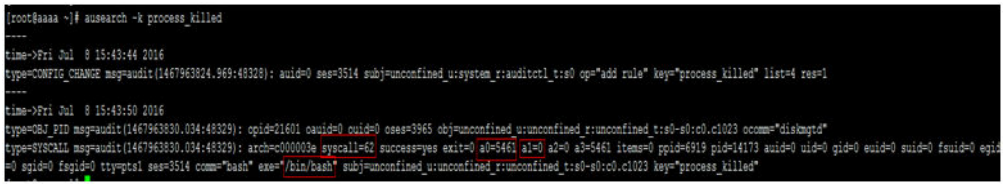
Add an audit rule:

auditctl -a exit,always -F arch=b64 -S kill -S tkill -S tgkill -F a1!=0 -k process_killed

View the rule:

auditctl -l

Step 3 If a process is killed due to an exception, you can run the **ausearch -k process_killed** command to query the kill history.



NOTE

a0 is the PID (hexadecimal) of the process that is killed, and **a1** is the semaphore of the kill command.

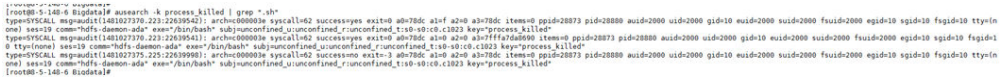
----End

Verification

Step 1 Restart an instance of the node on MRS Manager, for example, DataNode.

Step 2 Run the **ausearch -k process_killed** command to check whether logs are printed.

The following is an example of the **ausearch -k process_killed |grep ".sh"** command. The command output indicates that the **hdfs-daemon-ada*** script closed the DataNode process.



----End

Stop auditing the **kill** command.

Step 1 Run the **service auditd restart** command. The temporarily added kill command audit logs are cleared automatically.

Step 2 Run the **auditctl -l** command. If no information about killing a process is returned, the rule is cleared successfully.

----End

16.2.10 Network Is Unreachable When Using pip3 to Install the Python Package in an MRS Cluster

Issue

When the Python package is installed using pip3, an error message is displayed, indicating that the network is unreachable.

Symptom

When a user runs the pip3 install command to install the Python package, an error message is displayed, indicating that the network is unreachable. For details, see the following figure:

Cause Analysis

The customer does not bind an EIP to the Master node.

Procedure

Step 1 Log in to the MRS management console.

Step 2 Choose **Clusters** > **Active Clusters**, select the faulty cluster, and click its name to check the **Basic Information** on the **Dashboard** tab page.

Step 3 On the **Nodes** tab page, click the name of a Master node in the Master node group to log in to the ECS management console.

Step 4 Click the **EIPs** tab and click **Bind EIP** to bind an EIP to the ECS.

Step 5 Log in to the Master node and run the **pip3 install** command to install the Python package.

----End

16.2.11 Failed to Download the MRS Cluster Client

Issue

On the local Master host, a user attempts to download an MRS cluster client for another remote host. However, the system displays a message indicating that the network or parameter is abnormal.

Symptom

On the local Master host, a user attempts to download an MRS cluster client for another remote host. However, the system displays a message indicating that the network or parameter is abnormal.

Cause Analysis

- The two hosts are in different VPCs.
- The password is incorrect.
- The firewall is enabled on the remote host.

Procedure

- The two hosts are in different VPCs.

  Enable port 22 of the remote host.
- The password is incorrect.

  Check whether the password is correct. The password cannot contain special characters.
- The firewall is enabled on the remote host.

  Download the MRS cluster client to the server and run the **scp** command provided by Linux to remotely send the client to the remote host.

16.2.12 Failed to Scale Out an MRS Cluster

Issue

The MRS console is accessible and functions properly, but the MRS cluster fails to be scaled out.

Symptom

The MRS console is normal, and no alarm or error message is displayed on MRS Manager. However, an error message is displayed during cluster scale-out, indicating that the MRS cluster contains nodes that are not running.

Cause Analysis

An MRS cluster can be scaled in or out only when it is running properly. According to the error message, the possible cause is that the cluster status in the database is abnormal or is not updated. As a result, the nodes in the cluster are not in the running state.

Procedure

Step 1 Log in to the MRS console and click the cluster name to go to the cluster details page. Check that the cluster is in the **Running** state.

Step 2 Click **Nodes** to view the status of all nodes. Ensure that all nodes are in the **Running** state.

Step 3 Log in to the podMaster node in the cluster, switch to the MRS deployer node, and view the **api-gateway.log** file.

1. Run the **kubectl get pod -n mrs** command to view the **pod** of the MRS deployer node.
2. Run the **kubectl exec -ti ${***Pod of the deployer node***} -n mrs /bin/bash** command to log in to the pod. For example, run the **kubectl exec -ti mrsdeployer-78bc8c76cf-mn9ss -n mrs /bin/bash** command to access the deployer container of MRS.
3. In the **/opt/cloud/logs/apigateway** directory, view the latest **api-gateway.log** file and search for the required keyword (such as **ERROR**, **scaling**, **clusterScaling**, **HostState**, **state-check**, or the cluster ID) in the file to check the error type.
4. Rectify the fault based on the error information and perform the scale-out again.
   - If the scale-out is successful, no further action is required.
   - If the scale-out fails, go to **Step 4**.

Step 4 Run the **/opt/cloud/mysql -u${***Username***} -P${***Port***} -h${***Address***} -p${***Password***}** command to log in to the database.

Step 5 Run the **select cluster_state from cluster_detail where cluster_id=***Cluster ID***";** command to check the value of **cluster_state**.

- If the value of **cluster_state** is **2**, the cluster status is normal. Go to **Step 6**.
- If the value of **cluster_state** is not **2**, the cluster status in the database is abnormal. You can run the **update cluster_detail set cluster_state=2 where cluster_id="***Cluster ID***";** command to update the cluster status and then check the value of **cluster_state**.
  - If the value of **cluster_state** is **2**, the cluster status is normal. Go to **Step 6**.
  - If the value of **cluster_state** is not **2**, contact technical support.

Step 6 Run the **select host_status from host where cluster_di="***Cluster ID***";** command to query the cluster host status.

- If the host is in the started state, no further action is required.
- If the host is not in the started state, run the **update host set host_status='started' where cluster_id="***Cluster ID***";** command to update the host status to the database.
  - If the host is in the started state, no further action is required.
  - If the host is not in the started state, contact technical support.

----End

16.2.13 Error Occurs When MRS Executes the Insert Command Using Beeline

Issue

An error occurs when MRS executes the insert command using Beeline.

Symptom

When the **insert into** statement is executed in Beeline of Hive, the following error is reported:

```
Mapping run in Tez on Hive transactional table fails when data volume is high with error: "org.apache.hadoop.hive.ql.lockmgr.LockException Reason: Transaction... already aborted, Hive SQL state [42000]."
```

Cause Analysis

This problem is caused by improper cluster configuration and Tez resource setting.

Procedure

This problem can be solved by setting configuration parameters on Beeline.

Step 1 Set the following properties to optimize performance (you are advised to change them at the cluster level):

- Set **hive.auto.convert.sortmerge.join** to **true**.
- Set **hive.optimize.bucketmapjoin** to **true**.
- Set **hive.optimize.bucketmapjoin.sortedmerge** to **true**.

Step 2 Modify the following content to adjust the resources of Tez:

- Set **hive.tez.container.size** to the size of the Yarn container.
- Set **hive.tez.container.size** to the Yarn container size **yarn.scheduler.minimum-allocation-mb** or a smaller value (for example, a half or quarter of the Yarn container size). Ensure that the value does not exceed **yarn.scheduler.maximum-allocation-mb**.

----End

16.2.14 How Do I Upgrade EulerOS to Fix Vulnerabilities in an MRS Cluster?

Issue

EulerOS has vulnerabilities at the underlying layer. This section describes how to upgrade the OS to fix vulnerabilities for an MRS cluster.

Symptom

When the NSFOCUS software is used to test the cluster, vulnerabilities are found at the underlying layer in the EulerOS.

Cause Analysis

When the NSFOCUS software is used to test the cluster, it is found that vulnerabilities exist at the underlying layer in the EulerOS. The MRS service is deployed in the EulerOS. Therefore, the system needs to be upgraded to fix the vulnerabilities.

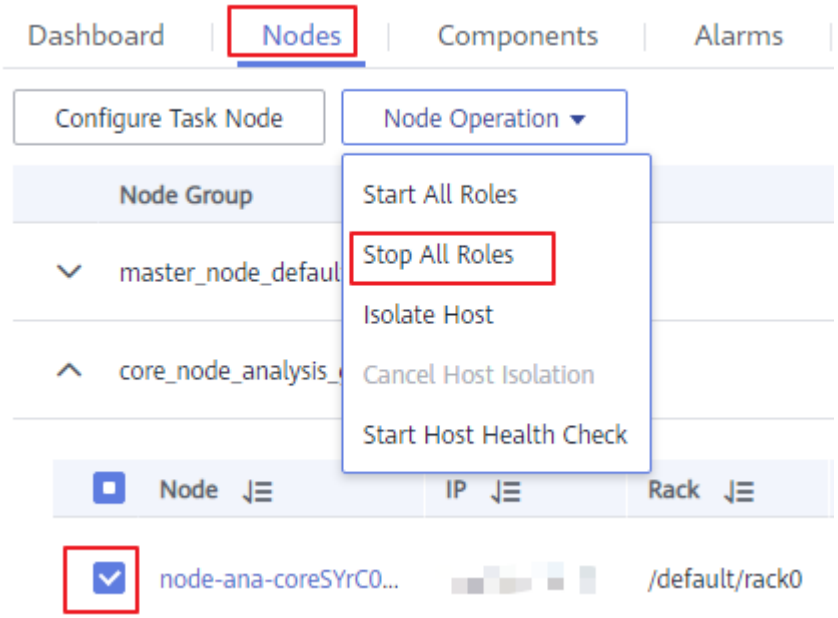
Procedure

> **NOTE**
>
> Before fixing the vulnerability, check whether Host Security Service (HSS) is enabled. If yes, disable HSS from monitoring the MRS cluster. After the vulnerability is fixed, enable HSS again.

Step 1 Log in to the MRS console.

Step 2 Click the cluster name. On the cluster details page, click the **Nodes** tab.

Step 3 In the core node group, select a core node, click **Node Operation**, and select **Stop All Roles**.



Step 4 Remotely log in to the core node and configure the yum repository.

Step 5 Run the **uname –r** or **rpm –qa |grep kernel** command to query and record the kernel version of the current node.

Step 6 Run the **yum update –y --skip-broken --setopt=protected_multilib=false** command to update the patch.

Step 7 After the update is complete, query the kernel version and run the **rpm -e** *Old kernel version* command to delete the old kernel version.

Step 8 On the cluster details page, click the **Nodes** tab.

Step 9 In the core node group, click the name of the core node whose patch has been updated. The ECS console is displayed.

Step 10 In the upper right corner of the page, click **Restart** to restart the core node.

Step 11 On the **Nodes** tab of the cluster details page, select the core node, click **Node Operation**, and select **Start All Roles**.

Step 12 Repeat **Step 1** to **Step 11** to upgrade other core nodes.

Step 13 After all core nodes are upgraded, upgrade the standby master node and then the active master node. For details, see **Step 1** to **Step 11**.

----End

16.2.15 Using CDM to Migrate Data to HDFS

Issue

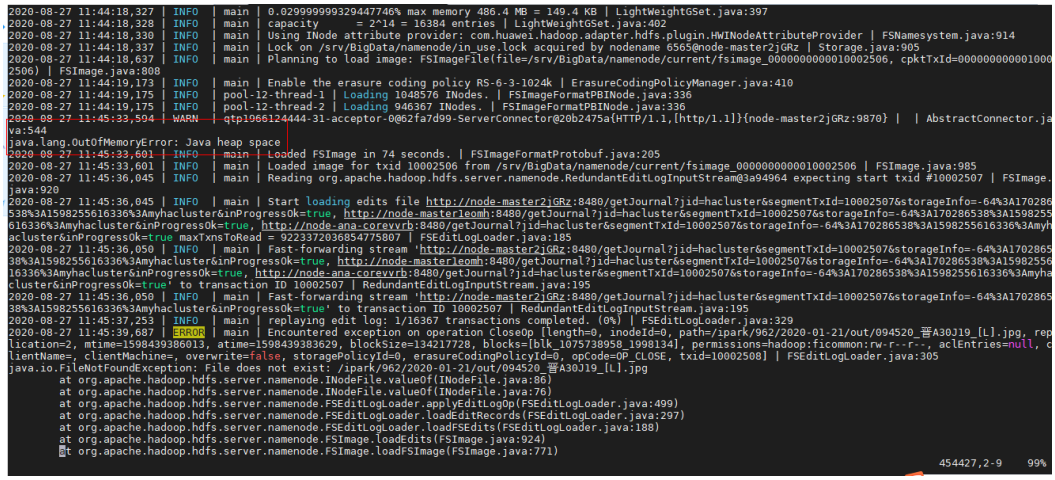
A user failed to use CDM to migrate data from an old cluster to HDFS of a new cluster.

Symptom

When CDM is used to import data from the source HDFS to the destination HDFS, the destination MRS cluster is faulty and the NameNode cannot be started.

The logs show that the **Java heap space** error is reported during the startup. The JVM parameter of the NameNode needs to be modified.

Figure 16-6 Fault logs



Cause Analysis

When the user uses CDM to migrate data, the HDFS data volume is too large. As a result, a stack exception occurs when metadata is merged.

Procedure

Step 1 Search for the **GC_OPTS** parameter in **HDFS->NameNode** and increase the values of **-Xms512M** and **-Xmx512M** based on service requirements.

Step 2 Save the configuration and restart the affected services or instances.

----End

16.2.16 Alarms Are Frequently Generated in the MRS Cluster

Issue

The cluster frequently reports alarms indicating that the heartbeat between the active and standby Manager nodes is interrupted, the heartbeat between the active and standby DBService nodes is interrupted, and the node is faulty. As a result, Hive is occasionally unavailable.

Symptom

The cluster frequently reports alarms indicating that the heartbeat between the active and standby Manager nodes is interrupted, the heartbeat between the active and standby DBService nodes is interrupted, and the node is faulty. As a result, Hive is occasionally unavailable, affecting customer services

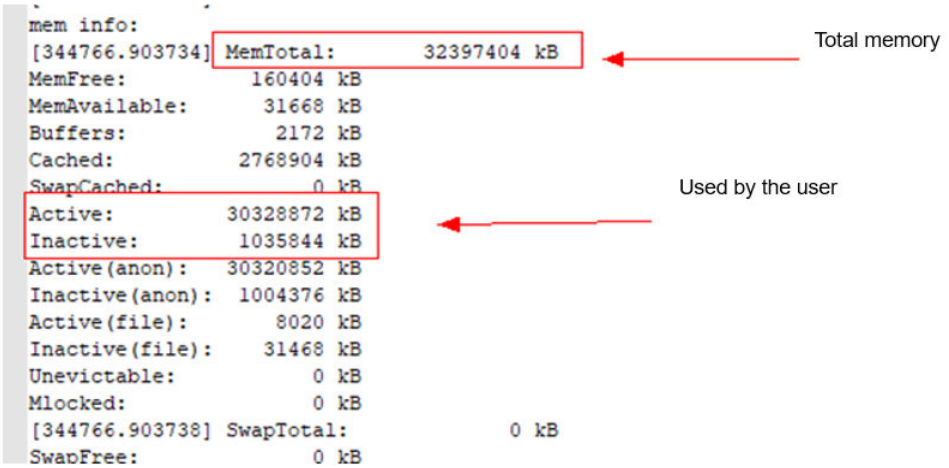
Cause Analysis

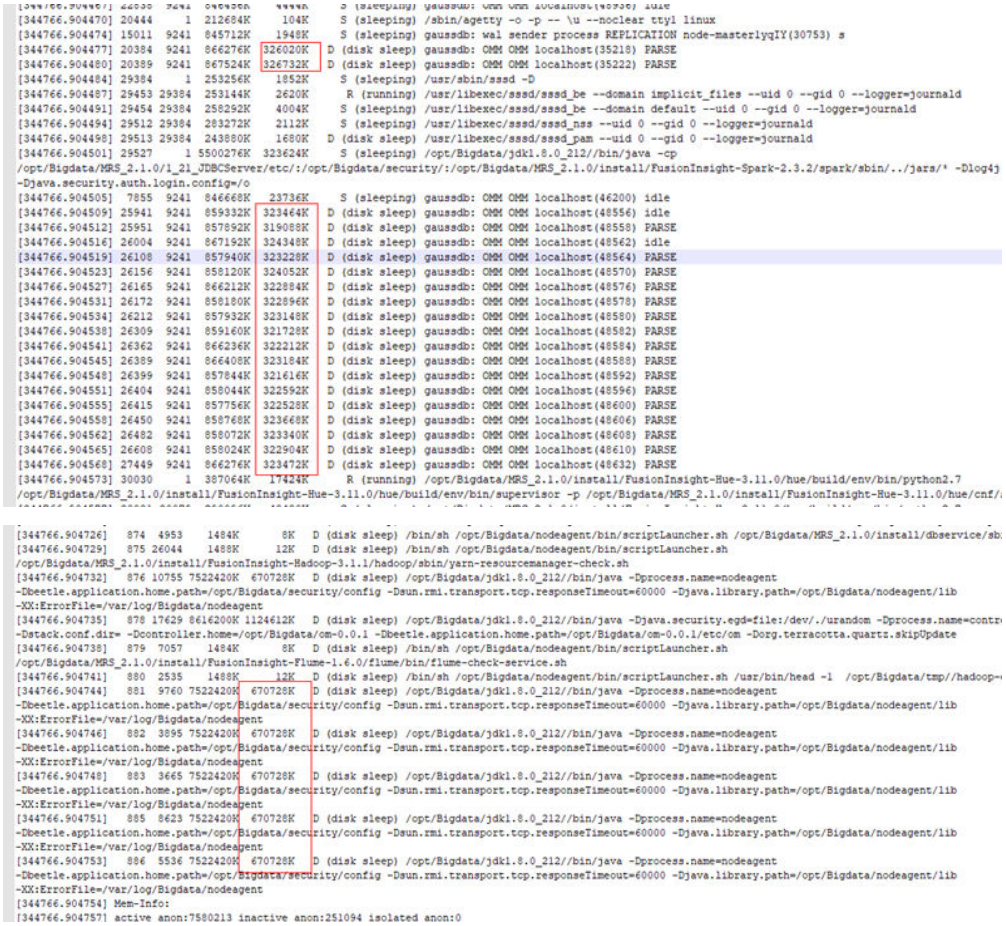
1. When the alarm is generated, the VM is restarted. The alarm is generated because the VM is restarted.

```
[omm@node-master1yqIY nodeagent]$ last
omm      pts/0        100.125.0.70     Thu Sep 24 10:33   still logged in
omm      pts/1        100.125.0.70     Thu Sep 24 09:26 - 09:47  (00:20)
omm      pts/0        100.125.0.70     Thu Sep 24 09:22 - 10:21  (00:59)
omm      pts/1        100.125.0.70     Wed Sep 23 17:32 - 17:37  (00:05)
root     pts/0        10.203.216.102   Wed Sep 23 17:13 - 18:35  (01:21)
omm      pts/0        100.125.0.70     Wed Sep 23 16:55 - 16:56  (00:00)
omm      pts/0        100.125.0.70     Wed Sep 23 16:20 - 16:25  (00:05)
reboot   system boot  4.19.36-vhulk190 Wed Sep 23 16:06   still running
root     pts/1        10.203.216.102   Tue Sep 22 19:13 - 19:48  (00:34)
omm      pts/0        100.125.0.70     Tue Sep 22 19:08 - 20:03  (00:54)
root     pts/0        10.203.216.102   Tue Sep 22 17:03 - 17:52  (00:48)
omm      pts/1        100.125.0.70     Tue Sep 22 15:55 - 16:00  (00:05)
```

```
[omm@node-master2WbYp ~]$ last
omm      pts/0        10.80.0.56       Thu Sep 24 11:00   still logged in
omm      pts/0        10.80.0.56       Thu Sep 24 09:24 - 10:21  (00:56)
omm      pts/0        10.80.0.56       Wed Sep 23 17:32 - 17:37  (00:05)
omm      pts/0        10.80.0.56       Tue Sep 22 19:15 - 19:15  (00:00)
omm      pts/0        10.80.0.56       Tue Sep 22 15:57 - 16:21  (00:23)
omm      pts/0        10.80.0.56       Tue Sep 22 15:23 - 15:35  (00:12)
omm      pts/0        10.80.0.56       Tue Sep 22 15:07 - 15:12  (00:05)
omm      pts/0        10.80.0.56       Tue Sep 22 14:21 - 14:26  (00:05)
omm      pts/0        10.80.0.56       Mon Sep 21 10:57 - 11:06  (00:09)
omm      pts/0        10.80.0.56       Mon Sep 21 10:42 - 10:56  (00:14)
omm      pts/0        10.80.0.56       Thu Sep 17 16:05 - 16:15  (00:10)
omm      pts/0        10.80.0.56       Wed Sep 16 20:52 - 20:58  (00:06)
reboot   system boot  4.19.36-vhulk190 Wed Sep 16 18:05   still running
omm      pts/0        10.80.0.56       Wed Sep 16 15:43 - 16:10  (00:26)
omm      pts/0        10.80.0.56       Wed Sep 16 14:35 - 14:53  (00:17)
omm      pts/0        10.80.0.56       Wed Sep 16 14:33 - 14:33  (00:00)
omm      pts/0        10.80.0.56       Wed Sep 16 14:11 - 14:29  (00:17)
omm      pts/0        10.80.0.56       Wed Sep 16 14:02 - 14:09  (00:06)
omm      pts/0        10.80.0.56       Wed Sep 16 11:56 - 12:04  (00:08)
omm      pts/0        10.80.0.56       Wed Sep 16 11:26 - 11:31  (00:04)
omm      pts/0        10.80.0.56       Wed Sep 16 11:09 - 11:24  (00:15)
root     pts/0        10.203.230.193   Mon Sep 14 15:54 - 16:30  (00:35)
root     pts/0        10.203.172.29    Fri Sep 11 17:15 - 17:45  (00:30)
root     pts/0        10.203.172.29    Fri Sep 11 16:53 - 17:12  (00:19)
root     tty1                          Fri Sep 11 16:23 - 17:25  (01:01)
reboot   system boot  4.19.36-vhulk190 Fri Sep 11 10:07   still running
reboot   system boot  4.19.36-vhulk190 Thu Aug 27 16:41   still running
root     tty1                          Thu Aug 20 09:46 - 10:17  (00:30)
reboot   system boot  4.19.36-vhulk190 Wed Aug 19 17:48   still running
reboot   system boot  4.19.36-vhulk190 Wed Aug 19 17:46   still running
```

2. According to the OS analysis, the cause of the VM restart is that the node does not have available memory. Memory overflow triggers oom-killer. When the process is invoked, the process enters the **disk sleep** state. As a result, the VM restarts.





3. Check the processes that occupy the memory. It is found that the processes that occupy the memory are normal service processes.

Conclusion: The VM memory cannot meet service requirements.

Procedure

- You are advised to expand the node memory.
- You are advised to disable unnecessary services to avoid this problem.

16.2.17 Memory Usage of the PMS Process Is High

Issue

What can I do if the memory usage of the active Master node is high?

Symptom

The memory usage of the active Master node is high. The **top -c** command output shows that the following idle processes occupy a large amount of memory:

```
12180 ommdba    20   0 1395492 1.180g 1.082g S   0.0  3.8  23:14.29 gaussdb: OMM OMM localhost(60598) idle
14828 ommdba    20   0 1395904 1.180g 1.081g S   0.0  3.8  23:17.08 gaussdb: OMM OMM localhost(60698) idle
15016 ommdba    20   0 1395840 1.180g 1.081g S   0.0  3.8  23:11.19 gaussdb: OMM OMM localhost(60824) idle
14943 ommdba    20   0 1395900 1.180g 1.081g S   0.0  3.8  23:14.76 gaussdb: OMM OMM localhost(60764) idle
14908 ommdba    20   0 1395840 1.180g 1.081g S   0.0  3.8  23:15.18 gaussdb: OMM OMM localhost(60738) idle
14953 ommdba    20   0 1395824 1.180g 1.081g S   0.0  3.8  23:15.96 gaussdb: OMM OMM localhost(60770) idle
14995 ommdba    20   0 1395560 1.180g 1.081g S   0.0  3.8  23:13.28 gaussdb: OMM OMM localhost(60812) idle
15062 ommdba    20   0 1395820 1.180g 1.081g S   0.0  3.8  23:16.12 gaussdb: OMM OMM localhost(60868) idle
15064 ommdba    20   0 1395512 1.180g 1.081g S   0.0  3.8  23:13.33 gaussdb: OMM OMM localhost(60870) idle
14973 ommdba    20   0 1395528 1.180g 1.081g S   0.0  3.8  23:12.74 gaussdb: OMM OMM localhost(60790) idle
14835 ommdba    20   0 1395536 1.180g 1.081g S   0.0  3.8  23:17.39 gaussdb: OMM OMM localhost(60704) idle
14822 ommdba    20   0 1395524 1.180g 1.081g S   0.0  3.8  23:13.80 gaussdb: OMM OMM localhost(60692) idle
14991 ommdba    20   0 1395808 1.180g 1.081g S   0.0  3.8  23:17.96 gaussdb: OMM OMM localhost(60808) idle
14975 ommdba    20   0 1395812 1.180g 1.081g S   0.0  3.8  23:12.57 gaussdb: OMM OMM localhost(60792) idle
15038 ommdba    20   0 1395520 1.180g 1.081g S   0.0  3.8  23:12.75 gaussdb: OMM OMM localhost(60846) idle
14919 ommdba    20   0 1395540 1.180g 1.081g S   0.0  3.8  23:11.58 gaussdb: OMM OMM localhost(60744) idle
14832 ommdba    20   0 1395476 1.180g 1.081g S   0.0  3.8  23:13.11 gaussdb: OMM OMM localhost(60702) idle
14989 ommdba    20   0 1395500 1.180g 1.081g S   0.0  3.8  23:15.63 gaussdb: OMM OMM localhost(60806) idle
14979 ommdba    20   0 1395448 1.180g 1.081g S   0.0  3.8  23:13.17 gaussdb: OMM OMM localhost(60796) idle
15047 ommdba    20   0 1395512 1.180g 1.081g S   0.0  3.8  23:12.10 gaussdb: OMM OMM localhost(60854) idle
14977 ommdba    20   0 1395496 1.180g 1.081g S   0.0  3.8  23:16.90 gaussdb: OMM OMM localhost(60794) idle
15028 ommdba    20   0 1395800 1.180g 1.081g S   0.0  3.8  23:09.35 gaussdb: OMM OMM localhost(60836) idle
```

Cause Analysis

- PostgreSQL cache: In addition to common execution plan cache and data cache, PostgreSQL provides cache mechanisms such as **catalog** and **relation** to improve the efficiency of generating execution plans. In the persistent connection scenario, some of the caches are not released. As a result, the persistent connection may occupy a large amount of memory.
- PMS is a monitoring process of MRS. This process frequently creates table partitions or new tables. The PostgreSQL caches the metadata of the objects accessed by the current session, and the connections in the database connection pool of the PMS exist for a long time. Therefore, the memory occupied by the connections gradually increases.

Procedure

Step 1 Log in to the active Master node as user **root**.

Step 2 Run the following command to query the PMS process ID:

ps -ef | grep =pmsd |grep -v grep

Step 3 Run the following command to stop the PMS process. In the command, **PID** indicates the PMS process ID obtained in **Step 2**.

kill -9 PID

Step 4 Wait for the PMS process to automatically start.

It takes 2 to 3 minutes to start PMS. PMS is a monitoring process. Restarting PMS does not affect big data services.

----End

16.2.18 High Memory Usage of the Knox Process

Issue

The memory usage of the knox process is high.

Symptom

The memory usage of the active Master node is high. The **top -c** command output shows that the memory usage of the knox process exceeds 4 GB.

Cause Analysis

The memory is not separately configured for the knox process. The process automatically allocates available memory based on the system memory size. As a result, the knox process occupies a large amount of memory.

Procedure

Step 1 Log in to the Master nodes as user **root**.

Step 2 Open the **/opt/knox/bin/gateway.sh** file. Search for **APP_MEM_OPTS**, and set its value to **-Xms3072m -Xmx4096m**.

Step 3 Log in to Manager and click **Hosts**. Find the IP address of the active Master node (that is, the node with a solid star before the hostname), and log in to the background of the node.

Step 4 Run the following commands to restart the process:

su - omm

sh /opt/knox/bin/restart-knox.sh

----End

16.2.19 It Takes a Long Time to Access HBase from a Client Installed on a Node Outside the Security Cluster

Issue

The cluster client is installed on a node outside the security cluster. When a user runs the **hbase shell** command on the client to access HBase, it is found that the access is very slow.

Symptom

A user creates a security cluster, installs a cluster client on a node outside the cluster, and runs the **hbase shell** command to access HBase. It is found that the access to HBase is very slow.

Cause Analysis

Kerberos authentication is required for a security cluster. You need to configure the **hosts** file on the client node to ensure that the access speed is not affected. An example of the **hosts** configuration is as follows:

```
1.1.1.1 hadoop.782670e3_1364_47e2_8c70_1b61bb80479c.com
1.1.1.1 hadoop.hadoop.com
1.1.1.1 hacluster
1.1.1.1 haclusterX
1.1.1.1 haclusterX1
```

```
1.1.1.1 haclusterX2
1.1.1.1 haclusterX3
1.1.1.1 haclusterX4
1.1.1.1 ClusterX
1.1.1.1 manager
ip1 hostname1
ip2 hostname2
ip3 hostname3
ip4 hostname4
```

Procedure

Copy the content of the **hosts** file on the cluster node to the **hosts** file on the node where the client is installed.

16.2.20 How Do I Locate a Job Submission Failure?

Symptom

A user cannot submit jobs through DGC or on the MRS console.

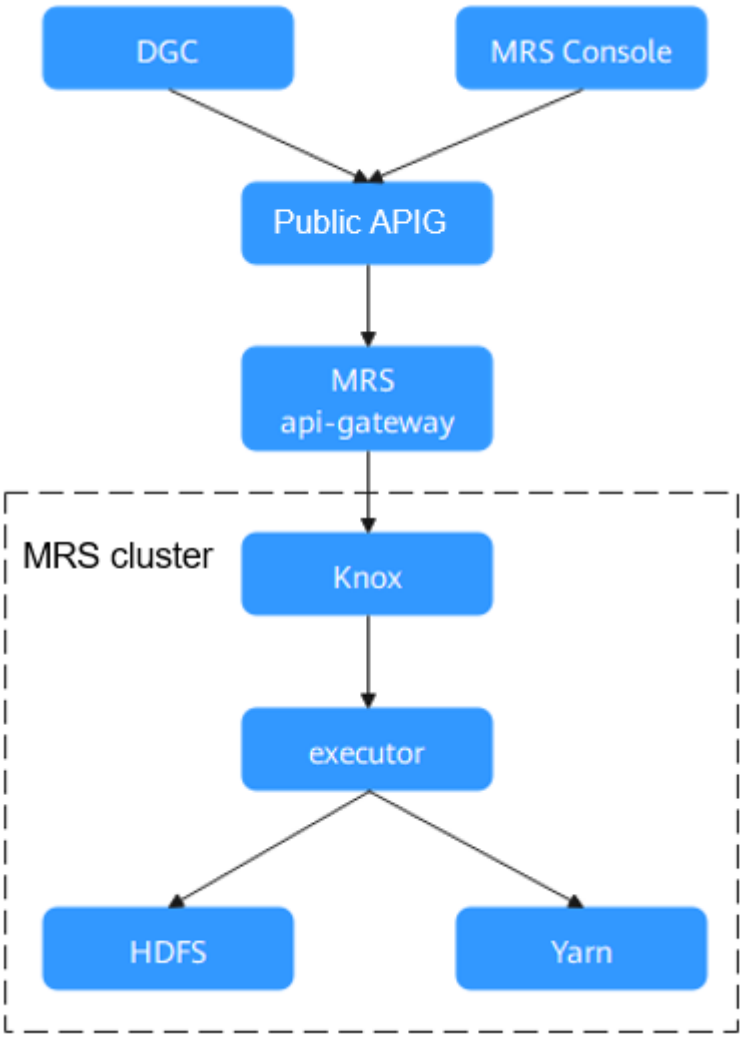
Impact

Jobs cannot be submitted, and services are interrupted.

Introduction to the Operation Process

1. All requests pass through APIG gateway and are restricted by the flow control configured on APIG.
2. APIG forwards the request to the api-gateway of the MRS management plane.
3. The API node on the MRS management plane polls the Knox of the active and standby OMS nodes to determine the Knox of the active OMS node.
4. MRS management-plane API submits a task to Knox of the active OMS.
5. Knox forwards requests to the Executor process on the current node.
6. The executor process submits a task to Yarn.

Figure 16-7 Job process



Procedure

Make preparations:

- Check whether the job is submitted through DGC or on the MRS console.
- Prepare the information listed in **Table 16-1**.

Table 16-1 Items to be prepared before the rectification

No.	Projects	Operation Mode
1	Cluster account information	Apply for **password** of user **admin** in the cluster.
2	Node account information	Apply for the passwords of users **omm** and **root** of cluster nodes.

No.	Projects	Operation Mode
3	Secure Shell (SSH) remote login tool	Prepare such tools as PuTTY or SecureCRT.
4	Client	Install the client.

Step 1 Locate the cause of the exception.

View the error code received in the job log and check whether the error code belongs to APIG or MRS.

- If the error code is a public APIG error code (starting with "APIGW"), contact public APIG maintenance personnel.
- If an error occurs on MRS, go to the next step.

Step 2 Check the running status of services and processes.

1. Log in to Manager and check whether a service fault occurs. If a job-related service fault or an underlying basic service fault occurs, rectify the fault.
2. Check whether a critical alarm is generated.
3. Log in to the active Master node.
4. Run the following command to check whether the OMS status is normal and whether the executor and knox processes on the active OMS node are normal: The knox is in active-active mode, and the executor is in single-active mode.

   /opt/Bigdata/om-0.0.1/sbin/status-oms.sh

5. Run the **jmap -heap** *PID* command as user **omm** to check the memory usage of the Knox and Executor processes. If the old-generation memory usage is 99.9%, the memory overflow occurs.

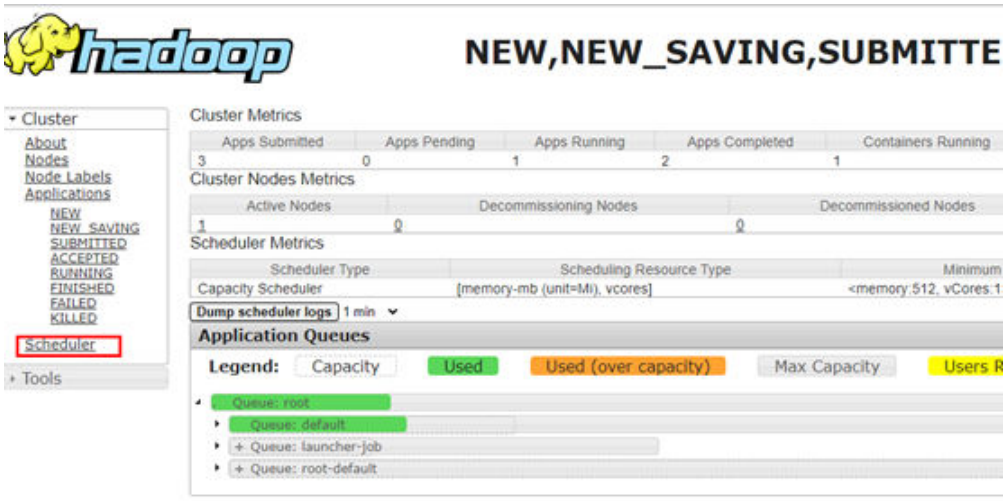
   Run the **netstat -anp | grep 8181 | grep LISTEN** command to query the PID of the executor process.

   Run the **ps -ef|grep knox | grep -v grep** command to query the PID of the knox process.

   If the memory overflows, run the **jmap -dump:format=b,file=/home/omm/temp.bin** *PID* command to export the memory information and restart the process.
6. View the native Yarn page to check the queue resource usage and whether the task is submitted to Yarn.

   On the native Yarn page: choose **Components** > **Yarn** > **ResourceManager WebUI** > **ResourceManager (Active)**.

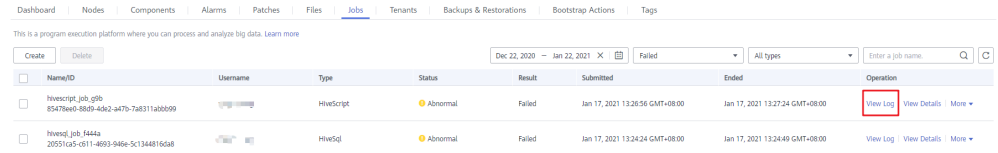
Figure 16-8 Queue resource usage on the Yarn page



Step 3 Locate the fault causing the task submission failure.

1. Log in to the MRS management console and click the cluster name to go to the cluster details page.
2. On the **Jobs** tab page, locate the row that contains the target job and click **View Log** in the **Operation** column.

   Figure 16-9 View the logs

3. If there is no log or the log information is not detailed, copy the job ID in the **Name/ID** column.
4. Run the following command on the active OMS node to check whether the task request is sent to the KNOX. If the request is not sent to the KNOX, the KNOX may be faulty. In this case, restart the KNOX to rectify the fault.

   grep "mrsjob" /var/log/Bigdata/knox/logs/gateway-audit.log | tail -10
5. Search for the job ID in the Executor log and view the error information.

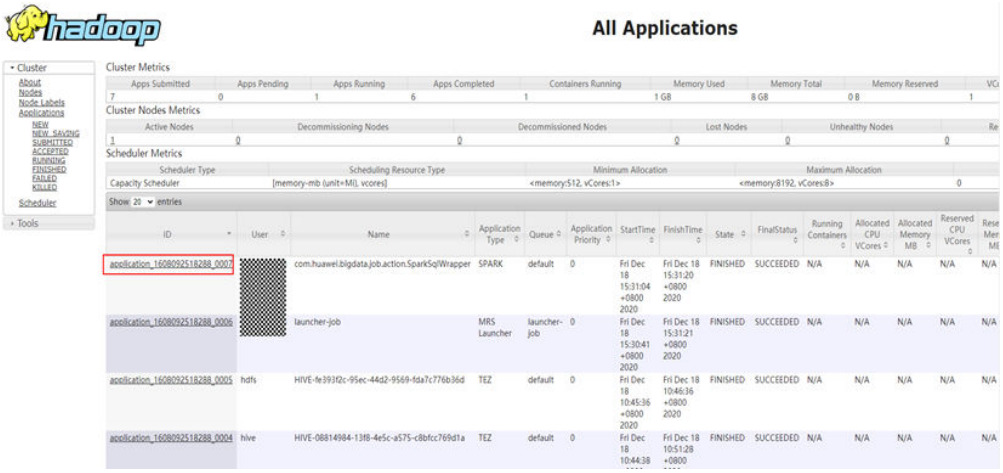
   Log file path: **/var/log/Bigdata/executor/logs/exe.log**
6. Modify the **/opt/executor/webapps/executor/WEB-INF/classes/log4j.properties** file to enable the debug log of the executor. Submit the test task and view the executor log. Confirm the error reported during job submission.

   Log file path: **/var/log/Bigdata/executor/logs/exe.log**
7. If an error occurs in the executor, run the following command to print the jstack information of the executor and check the current execution status of the thread:

   jstack PID > xxx.log
8. On the cluster details page, click the **Jobs** tab. Locate the row that contains the target job, and click **View Details** in the **Operation** column to obtain the actual job ID (**applicationID**).

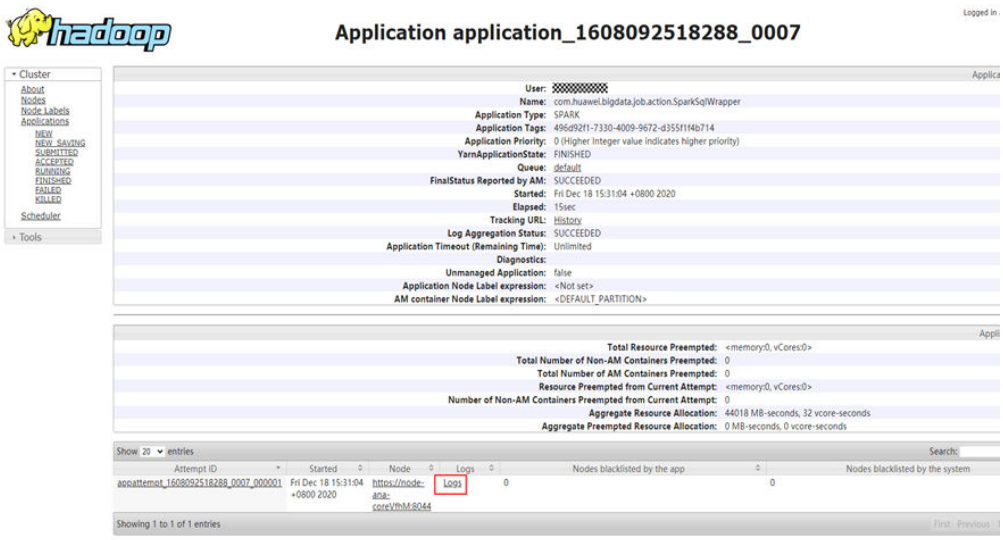
9. On the cluster details page, choose **Components** > **Yarn** > **ResourceManager WebUI** > **ResourceManager (Active)**. On the native Yarn page that is displayed, click **applicationID**.

Figure 16-10 Yarn applications



10. View logs on the task details page.

Figure 16-11 Task logs



----End

16.2.21 OS Disk Space Is Insufficient Due to Oversized HBase Log Files

Issue

The space of the **/var/log** partition on the system disk is insufficient.

Symptom

The **/var/log/Bigdata/hbase/*/hbase-omm-*.out** log file is too large, causing insufficient space of the **/var/log** partition on the system disk.

Cause Analysis

During the long-term running of HBase, the OS periodically deletes the **/tmp/.java_pid*** files created by the JVM. The HBase memory monitoring uses the **jinfo** command, which depends on the **/tmp/.java_pid*** file. If the file does not exist, the **jinfo** command runs **kill -3** to print the jstack information to the **.out** log file. As a result, the **.out** log file becomes oversize as time goes by.

Procedure

On each node hosting the HBase instance, deploy a scheduled task to periodically clear the **.out** log file. For example, log in to the HBase instance node and run the **crontab -e** command to add a scheduled task to clear the **.out** log file at 00:00:00 every day.

crontab -e

00 00 * * * for file in `ls /var/log/Bigdata/hbase/*/hbase-omm-*.out`; do echo "" > $file; done

NOTE

If large **.out** files are generated frequently, you can clear the files multiple times every day or adjust the automatic clearing policy of the OS.

16.2.22 Failed to Delete a New Tenant on FusionInsight Manager

Symptom

A user fails to delete a tenant created on the **Tenant Resources** page of FusionInsight Manager, and an error message is displayed.

Cause Analysis

When a tenant is created, its role is generated. The role will be deleted first when the tenant is deleted. If the component that supports permission configuration is abnormal, the resource permission of the role fails to be deleted.

Procedure

Step 1 Log in to FusionInsight Manager and choose **System** > **Permission** > **Role**.

Step 2 Click **Create Role**. In the **Configure Resource Permission** area, click the cluster name to check the components available for resource permission configuration.

Step 3 Choose **Cluster** > **Services** and check that the running status of these components is **Normal**.

Step 4 (Optional) If the running status is not **Normal**, start or repair the component until its running status becomes **Normal**.

Step 5 Delete the tenant again.

----End

16.3 Using Alluixo

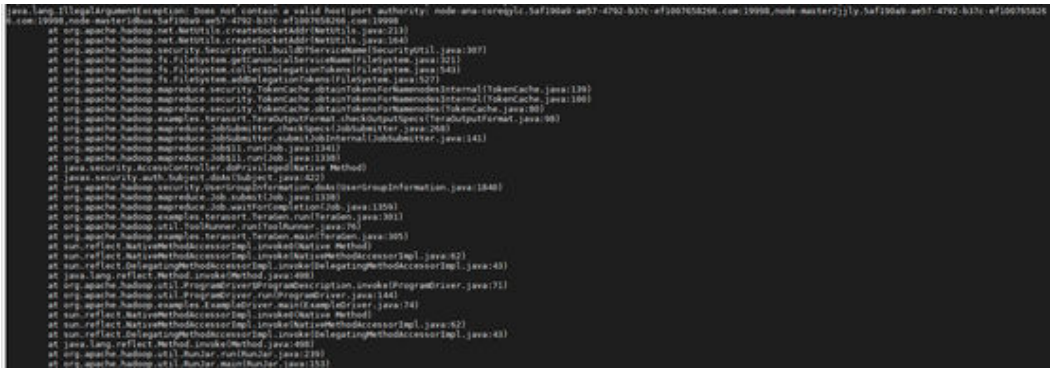
16.3.1 Error Message "Does not contain a valid host:port authority" Is Reported When Alluixo Is in HA Mode

Issue

Error message "Does not contain a valid host:port authority" is reported for Alluixo in HA mode in a security cluster.

Symptom

Error message "Does not contain a valid host:port authority" is reported for Alluixo in HA mode in a security cluster.



Cause Analysis

org.apache.hadoop.security.SecurityUtil.buildDTServiceName does not support multiple alluxiomaster addresses in the URI.

Procedure

Use **alluxio:///** or **alluxio://**<*IP address or hostname of the active AlluxioMaster*>**:19998/** for access.

16.4 Using ClickHouse

16.4.1 ClickHouse Fails to Start Due to Incorrect Data in ZooKeeper

Symptom

An instance node in the ClickHouse cluster fails to start. The startup log of the instance node contains error information similar to the following:

```
2021.03.15 21:01:19.816593 [ 11111 ] {} <Error> Application: DB::Exception:
The local set of parts of table DEFAULT.lineorder doesn't look like the set ofdoesn't look like the set of
parts in ZooKeeper: 59.99 million rows of 59.99 million total rows in
filesystem are suspicious. There are 30 unexpected parts with 59986052 rows
(14 of them is not just-written with 59986052 rows), 0 missing parts (with 0
blocks).: Cannot attach table `DEFAULT`.`lineorder` from metadata file
...
: while loading database
```

Cause Analysis

When a ClickHouse instance is abnormal, the ReplicatedMergeTree engine table is repeatedly created in the cluster, and then deleted. The creation and deletion of the ReplicatedMergeTree engine table causes data error in ZooKeeper, which causes a start failure of ClickHouse.

Solution

Step 1 Back up all data tables in the database of the faulty node to another directory.

- Back up table data:

  cd /srv/BigData/data*1*/**clickhouse/data/***Database name*

  mv *Table name Directory to be backed up/***data***1*

  NOTE

  If there are multiple disks, back up data of **data1** to **dataN**.

- Back up metadata information:

  cd /srv/BigData/data1/clickhouse_path/metadata

  mv *Table name***.sql** *Directory to be backed up*

For example, to back up the lineorder table in the default database to the **/home/backup** directory, run the following command.

cd /srv/BigData/data1/clickhouse/data/default

mv lineorder /home/backup/data1

cd /srv/BigData/data1/clickhouse_path/metadata

mv lineorder.sql /home/backup

Step 2 Log in to MRS Manager, choose **Cluster** > **Services** > **ClickHouse** > **Instance**, select the target instance node, and click **Start Instance**.

Step 3 After the instance is started, use the ClickHouse client to log in to the faulty node.

clickhouse client --host *Clickhouse instance IP address* **--user** *User name* **--password** *Password*

Step 4 Run the following command to obtain the ZooKeeper path **zookeeper_path** of the current table and **replica_num** of the corresponding node.

SELECT zookeeper_path FROM system.replicas WHERE database = '*Database name*' **AND table = '***Table name*'**;**

SELECT replica_num,host_name FROM system.clusters;

Step 5 Run the following command to access the ZooKeeper command line interface:

zkCli.sh -server *IP address of the ZooKeeper node*:**2181**

Step 6 Locate the ZooKeeper path corresponding to the table data of the faulty node.

ls *zookeeper_path*/**replicas/***replica_num*

NOTE

zookeeper_path indicates the value of **zookeeper_path** obtained in **Step 4**.

replica_num indicates the value of **replica_num** corresponding to the host in **Step 4**.

Step 7 Run the following command to delete the replica data from ZooKeeper:

deleteall *zookeeper_path*/**replicas/***replica_num*

Step 8 Use the ClickHouse client to log in to the node and create the ReplicatedMergeTree engine table of the cluster.

clickhouse client --host *Clickhouse instance IP address* **--multiline --user** *Username* **--password** *Password*

CREATE TABLE *Database name.Table name* **ON CLUSTER** *Cluster name*

...

ENGINE = ReplicatedMergeTree ...

The following error message is displayed on other replica nodes, which is normal and can be ignored.

```
Received exception from server (version 20.8.7):
Code: 57. DB::Exception: Received from x.x.x.x:9000. DB::Exception:
There was an error on [x.x.x.x:9000]: Code: 57, e.displayText() =
DB::Exception: Table DEFAULT.lineorder already exists. (version 20.8.11.17
(official build)).
```

After the table is successfully created, the table data on the faulty node will be automatically synchronized. The data restoration is complete.

----End

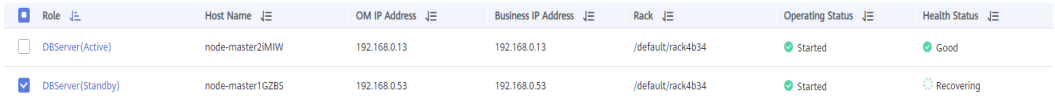
16.5 Using DBService

16.5.1 DBServer Instance Is in Abnormal Status

Symptom

A DBServer instance is in the **Concerning** state for a long period of time.

Figure 16-12 DBServer instance status



Role	Host Name	OM IP Address	Business IP Address	Rack	Operating Status	Health Status
DBServer(Active)	node-master2iMIW	192.168.0.13	192.168.0.13	/default/rack4b34	Started	Good
DBServer(Standby)	node-master1GZB5	192.168.0.53	192.168.0.53	/default/rack4b34	Started	Recovering

Cause Analysis

The permission for files or directories in the data directory is incorrect. GaussDB requires that the file permission be at least 600 and directory permission be at least 700.

Figure 16-13 Directory permission list

```
omm@192-168-234-176:/srv/BigData/dbdata_service> ll
total 4
drwx------ 19 omm wheel 4096 Dec 14 10:15 data
```

Figure 16-14 File permission list

```
omm@192-168-234-176:/srv/BigData/dbdata_service/data> ll
total 128
drwx------ 6 omm wheel   4096 Dec  9 15:47 base
-rw------- 1 omm wheel    922 Dec  9 15:34 dblink.conf
-rw------- 1 omm wheel     16 Dec 14 10:15 gaussdb.state
drwx------ 2 omm wheel   4096 Dec 14 10:17 global
drwx------ 2 omm wheel   4096 Dec 11 00:00 pg_audit
drwx------ 2 omm wheel   4096 Dec 14 10:15 pg_blackbox
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_clog
drwx------ 2 omm wheel   4096 Dec 14 10:15 pg_confile_backup
-rw------- 1 omm wheel   1024 Dec  9 15:34 pg_ctl.lock
-rw------- 1 omm wheel   4245 Dec  9 15:47 pg_hba.conf
-rw------- 1 omm wheel   1024 Dec  9 15:47 pg_hba.conf.lock
-rw------- 1 omm wheel   1636 Dec  9 15:34 pg_ident.conf
drwx------ 2 omm wheel   4096 Dec  9 15:38 pg_log
drwx------ 4 omm wheel   4096 Dec  9 15:34 pg_multixact
drwx------ 2 omm wheel   4096 Dec 14 10:15 pg_notify
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_serial
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_snapshots
drwx------ 2 omm wheel   4096 Dec 14 11:56 pg_stat_tmp
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_subtrans
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_tblspc
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_twophase
-rw------- 1 omm wheel      4 Dec  9 15:34 PG_VERSION
drwx------ 2 omm wheel   4096 Dec  9 15:34 pg_wallet
drwx------ 3 omm wheel   4096 Dec  9 15:39 pg_xlog
-rw------- 1 omm wheel  13309 Dec 14 10:15 postgresql.conf
-rw------- 1 omm wheel   1024 Dec  9 15:34 postgresql.conf.lock
-rw------- 1 omm wheel    105 Dec 14 10:15 postmaster.opts
-rw------- 1 omm wheel     96 Dec 14 10:15 postmaster.pid
```

Solution

Step 1 Modify the permissions on the files and directories based on the permission list in **Figure 16-13** and **Figure 16-14**.

Step 2 Restart the DBServer instance.

----End

16.5.2 DBServer Instance Remains in the Restoring State

Symptom

A DBServer instance remains in the **Restoring** state. The status cannot be recovered even after a restart.

Cause Analysis

1. DBService monitors the **${BIGDATA_HOME}/MRS_*XXX*/install/dbservice/ha/module/harm/plugin/script/gsDB/.startGS.fail** file. *XXX* indicates the product version.
2. If the value in the file is greater than 3, the startup fails. The NodeAgent keeps trying to restart the instance. In this case, the startup still fails and the value is incremented by 1 each time the startup fails.

Solution

Step 1 Log in to MRS Manager.

Step 2 Stop the DBServer instance.

Step 3 Log in to the node where the DBServer instance is abnormal as user **omm**.

Step 4 Change the value of in the **${BIGDATA_HOME}/MRS_*XXX*/install/dbservice/ha/module/harm/plugin/script/gsDB/.startGS.fail** file to **0**. *XXX* indicates the product version.

Step 5 Start the DBServer instance.

----End

16.5.3 Default Port 20050 or 20051 Is Occupied

Symptom

DBService restart fails, and information indicating that port 20050 or 20051 is occupied is displayed in the printed fault log.

Cause Analysis

1. The default port 20050 or 20051 used by DBService is occupied by another process.
2. The DBService process is not stopped, and the port used by DBService is not released.

Solution

This solution uses port 20051 as an example. The solution to the problem that port 20050 is occupied is similar.

Step 1 Log in to the node where the error is reported as user **root**, and run the **netstat -nap | grep 20051** command to check the process that occupies port 20051.

Step 2 Run the **kill** command to forcibly stop the process that uses port 20051.

Step 3 About 2 minutes later, run the **netstat -nap | grep 20051** command again to check whether any process uses the port.

Step 4 Check the service to which the process belongs and change the port for the service.

Step 5 Run the **find . -name "*20051*"** command in the **/tmp** and **/var/run/MRS-DBService/** directories, and delete all files found.

Step 6 Log in to Manager and restart DBService.

----End

16.5.4 DBServer Instance Is Always in the Restoring State Because the Incorrect /tmp Directory Permission

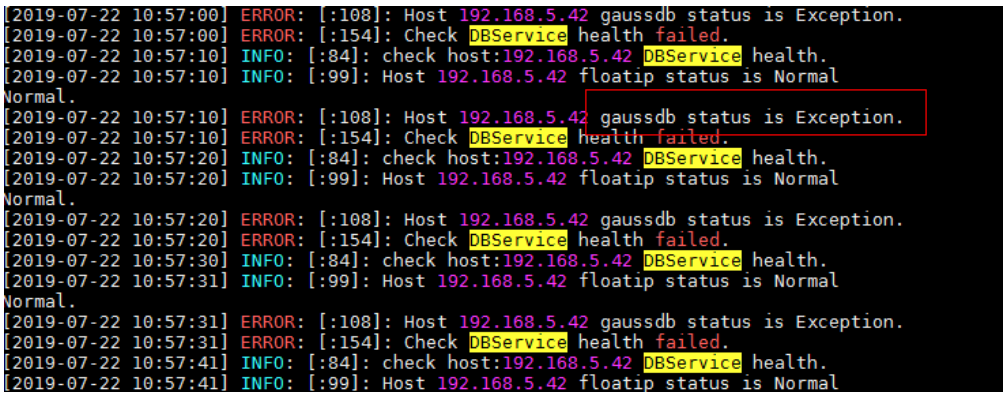
Symptom

A DBServer instance remains in the **Restoring** state. The status cannot be recovered even after a restart.

Cause Analysis

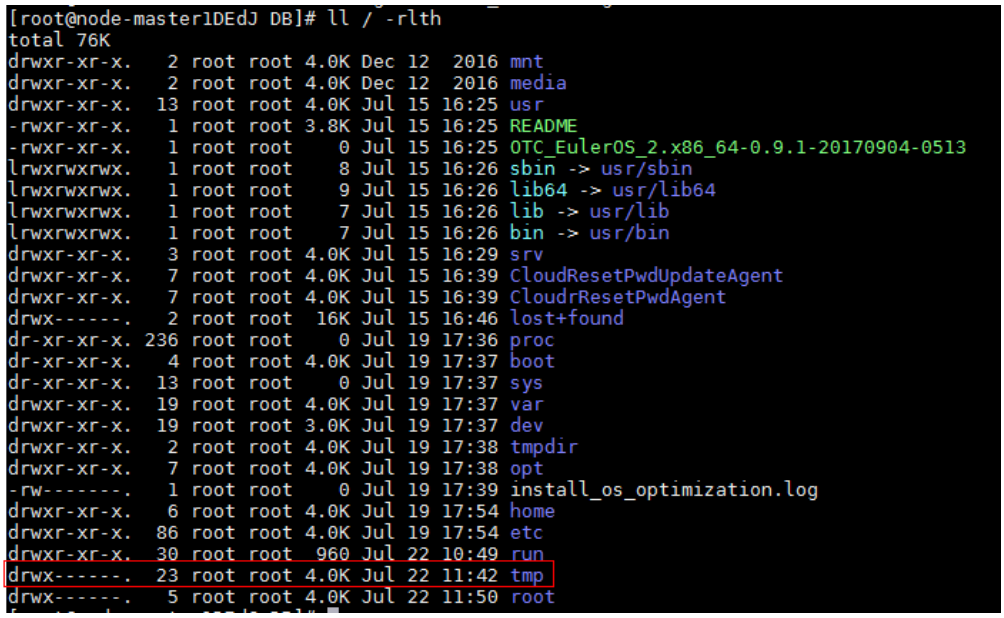
1. Check **/var/log/Bigdata/dbservice/healthCheck/dbservice_processCheck.log**. It is found that GaussDB is abnormal.

   Figure 16-15 GaussDB exception

2. The check result shows that the permission on the **/tmp** directory is incorrect.

Figure 16-16 /tmp permission



Solution

Step 1 Run the following command to modify the **/tmp** permission:

chmod 1777 /tmp

Step 2 Wait until the instance status recovers.

----End

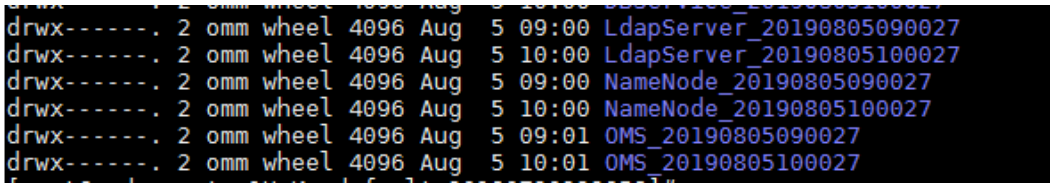
16.5.5 DBService Backup Failure

Symptom

ls /srv/BigData/LocalBackup/default_20190720222358/ -rlth

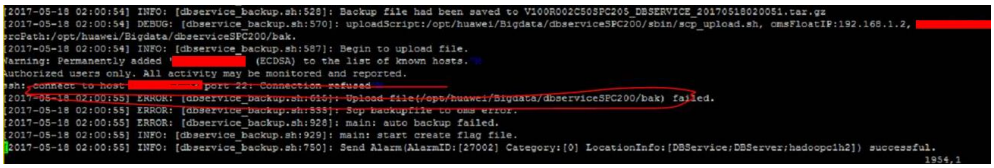
No DBService backup file exists in the backup file path.

Figure 16-17 Checking the backup file



Cause Analysis

- Check the backup log of DBService in **/var/log/Bigdata/dbservice/scriptlog/backup.log**. It is found that the backup is successful but fails to be uploaded to the OMS node.



- The failure is caused by the SSH failure.

Solution

Step 1 If the network is faulty, contact network engineers.

Step 2 Perform backup operations again after the network fault is rectified.

----End

16.5.6 Components Failed to Connect to DBService in Normal State

Symptom

Upper-layer components fail to connect to DBService. The DBService component and two instances are normal.

Figure 16-18 DBService status

Cause Analysis

1. The upper-layer component is DBService connected through **dbservice.floatip**.
2. Run the **netstat -anp | grep 20051** command on the node where DBServer resides. It is found that the Gauss process of DBService is not bound to the floating IP address during startup, and only the local IP address 127.0.0.1 is listened.

Solution

Step 1 Restart the DBService service.

Step 2 Run the **netstat -anp | grep 20051** command on the active DBServer node to check whether **dbservice.floatip** is bound.

----End

16.5.7 DBServer Failed to Start

Symptom

DBService fails to be started and restarts also fail. The instance keeps in the **Recovering** state.

Figure 16-19 DBService status

Role	Host Name	OM IP Address	Business IP Address	Rack	Operating Status	Health Status
DBServer(Active)	node-master2iMIW	192.168.0.13	192.168.0.13	/default/rack4b34	Started	Good
DBServer(Standby)	node-master1GZBS	192.168.0.53	192.168.0.53	/default/rack4b34	Started	Recovering

Cause Analysis

1. Check the DBService logs in **/var/log/Bigdata/dbservice/DB/gs_ctl-current.log**. The following error message is displayed:

```
OCATION:  PostmasterMain, postmaster.c:798
OG:  Starting SelectConfigFiles (postmaster.c:1049)
2017-09-23 15:19:03.591 CST]  gaussmaster 922216  LOG:  Starting checkDataDir (postmaster.c:1068)
2017-09-23 15:19:03.591 CST]  gaussmaster 922216  LOG:  Starting ChangeToDataDir (postmaster.c:1074)
2017-09-23 15:19:03.591 CST]  gaussmaster 922216  LOG:  Starting CheckDateTokenTables (postmaster.c:1120)
2017-09-23 15:19:03.591 CST]  gaussmaster 922216  LOG:  Starting CreateDataDirLockFile (postmaster.c:1151)
2017-09-23 15:19:03.596 CST]  gaussmaster 922216  LOG:  Starting pgaudit_agent_init (postmaster.c:1169)
2017-09-23 15:19:03.596 CST]  gaussmaster 922216  LOG:  Starting process_shared_preload_libraries (postmaster.c:1178)
2017-09-23 15:19:03.597 CST]  gaussmaster 922216  LOG:  could not bind IPv4 socket at the 0 time: ?????????? (pqcomm.c:562)
2017-09-23 15:19:03.597 CST]  gaussmaster 922216  HINT:  Is another postmaster already running on port 20051? If not, wait a few seconds and retry.
2017-09-23 15:19:03.698 CST]  gaussmaster 922216  LOG:  could not bind IPv4 socket at the 1 time: ?????????? (pqcomm.c:562)
2017-09-23 15:19:03.698 CST]  gaussmaster 922216  HINT:  Is another postmaster already running on port 20051? If not, wait a few seconds and retry.
2017-09-23 15:19:03.798 CST]  gaussmaster 922216  LOG:  could not bind IPv4 socket at the 2 time: ?????????? (pqcomm.c:562)
2017-09-23 15:19:03.798 CST]  gaussmaster 922216  HINT:  Is another postmaster already running on port 20051? If not, wait a few seconds and retry.
2017-09-23 15:19:03.898 CST]  gaussmaster 922216  WARNING:  could not create listen socket for "192.168.5.162" (postmaster.c:1235)
2017-09-23 15:19:03.898 CST]  gaussmaster 922216  LOG:  discard audit data: could not create lock file "/tmp/.s.PGSQL.20051.lock": ???? (pgaudit.c:1961)
2017-09-23 15:19:03.898 CST]  gaussmaster 922216  FATAL:  could not create lock file "/tmp/.s.PGSQL.20051.lock": ???? (miscinit.c:854)
```

2. It is found that the **/tmp** permission is incorrect. The correct value should be **777**.

```
mm@hadoopc1h2:/var/log/Bigdata/dbservice/DB> ll /
otal 100
rwxr-xr-x   2 root  root     4096 Aug  6  2016 bin
rwxr-xr-x   3 root  root     4096 Aug  6  2016 boot
rwxr-xr-x  17 root  root     5080 Sep 20 11:30 dev
rwxr-xr-x   3 httpd common      0 Sep 20 11:20 ecmramfs
rwxr-xr-x  71 root  root     4096 Sep 22 02:40 etc
rw-r-----   1 root  root        0 Sep 11 08:25 fsck_corrected_
rwxr-xr-x   9 root  root     4096 Sep 18 14:39 home
rwxr-xr-x  12 root  root     4096 Sep 14  2016 lib
rwxr-xr-x   8 root  root    12288 Sep 14  2016 lib64
rwx------   2 root  root    16384 Aug  7  2016 lost+found
rwxr-xr-x   2 root  root     4096 May  5  2010 media
rwxr-xr-x   2 root  root     4096 May  5  2010 mnt
rwxr-xr-x  19 root  root     4096 Jun 30 10:04 opt
r-xr-xr-x 424 root  root        0 Sep 20 19:18 proc
rwx------   5 root  root     4096 Sep 23 10:21 root
rwxrwxr-x   4 root  root     4096 Aug  7  2016 rrdtool
rwxr-xr-x   3 root  root    12288 Sep 14  2016 sbin
rwxr-xr-x   2 root  root     4096 May  5  2010 selinux
rwxrwxrwx  10 root  root     4096 Nov 15  2016 srv
rwxr-xr-x  12 root  root        0 Sep 20 11:19 sys
rwxrwxrwx   1 root  root        1 Aug  7  2016 target -> /
drwxr-xr-x   6 root  root     4096 Sep 23 15:19 tmp
drwxr-xr-x  13 root  root     4096 Apr 22  2014 usr
```

Solution

Step 1 Modify the **/tmp** permission by changing the value to **777**.

Step 2 Restart DBService.

----End

16.5.8 DBService Backup Failed Because the Floating IP Address Is Unreachable

Symptom

The default DBService backup fails, but backups of NameNode, LdapServer, and OMS are successful.

Cause Analysis

1. Check the error information on the DBService backup page:
   Clear temporary files at backup checkpoint DBService_test_DBService_DBService_20180326155921 that fialed last time.
   Temporyary files at backup checkpoint DBService_test_DBService_DBService20180326155921 that failed last time are cleared successfully.

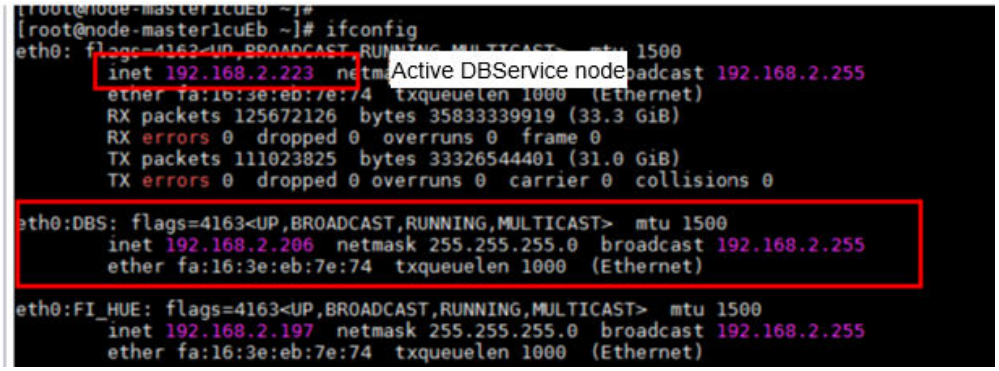
```
Start executing the backup task.
The backup of configuration DBService is started.
Check the backup available disk space.
Backup initialization succeeded for configuration DBService.
Clear temporary files at backup checkpoint DBService_test_DBService_DBService_20180326155921 that failed last time.
Temporary files at backup checkpoint DBService_test_DBService_DBService_20180326155921 that failed last time are cleared successfully.
Checkpoint DBService_test_DBService_DBService_20180326162235 is verified successfully before backup.
Temporary files are cleared successfully before backup checkpoint DBService_test_DBService_DBService_20180326162235.
Prestart backup succeeded for checkpoint DBService_test_DBService_DBService_20180326162235.
The snapshot is created successfully for checkpoint DBService_test_DBService_DBService_20180326162235 before backup.
Backup is being performed for checkpoint DBService_test_DBService_DBService_20180326162235.
Backup execution failed. Task ID: 2
Detail: DBService backup task failed, please view details in logs.
Temporary files are cleared successfully after backup checkpoint DBService_test_DBService_DBService_20180326162235.
checkpoint DBService_test_DBService_DBService_20180326162235 is deleted successfully after backup failure.
Failed to backup configuration DBService.
```

2. Check the **/var/log/Bigdata/dbservice/scriptlog/backup.log** file. It is found that the log printing stops and no related backup information is found.
3. Check the **/var/log/Bigdata/controller/backupplugin.log** file on the active OMS node. The following error information is found:
   result error is ssh:connect to host 172.16.4.200 port 22: Connection refused (172.16.4.200 is the floating IP address of DBService)
   DBService backup failed.

```
2018-03-27 07:00:35,758 INFO  [pool-1-thread-5] Create adapter from com.huawei.bigdata.om.backup.MetadataPluginAdapter success.
com.huawei.bigdata.om.backup.plugin.AbstractBackupRecoveryPlugin.initializePluginAdapter(AbstractBackupRecoveryPlugin.java:92)
2018-03-27 07:00:35,759 INFO  [pool-1-thread-5] floatIp is 172.16.4.200. com.huawei.bigdata.om.dbservice.backup.BackupRecoveryPlugin.getFloatIp(BackupRecoveryPlugin.java:233)
2018-03-27 07:00:35,759 INFO  [pool-1-thread-5] cmd is ssh 172.16.4.200 /opt/huawei/Bigdata/FusionInsight_V100R002C60U20/dbservice/sbin/dbservice_backup.sh -b -d
/srv/BigData/LocalBackup/default_20180326213206/DBService_20180327070010. com.huawei.bigdata.om.dbservice.backup.BackupRecoveryPlugin.startBackup(BackupRecoveryPlugin.java:166)
2018-03-27 07:00:35,759 INFO  [pool-1-thread-5] create task taskId is 6. com.huawei.bigdata.om.dbservice.backup.BackupRecoveryPlugin.startBackup(BackupRecoveryPlugin.java:169)
2018-03-27 07:00:35,760 INFO  [pool-1-thread-5] startBackup result OperateResult(errorCode:RUNNING, result:6, detailInfo:, packageName:null).
com.huawei.bigdata.om.backup.BackupPluginContainerHandler.startBackup(BackupPluginContainerHandler.java:246)
2018-03-27 07:00:35,760 INFO  [Thread-132] Executing the command with arguments and env, timeout: 900000
com.huawei.bigdata.om.controller.api.extern.monitor.script.LinuxScriptExecutionHandler.logMessage(LinuxScriptExecutionHandler.java:64)
2018-03-27 07:00:35,863 INFO  [Thread-132] Execute command : /opt/huawei/Bigdata/om-0.0.1/sbin/scriptLauncher.sh ssh 172.16.4.200
/opt/huawei/Bigdata/FusionInsight_V100R002C60U20/dbservice/sbin/dbservice_backup.sh -b -d /srv/BigData/LocalBackup/default_20180326213206/DBService_20180327070010.
com.huawei.bigdata.om.dbservice.backup.BackupTask.run(BackupTask.java:48)
2018-03-27 07:00:35,863 INFO  [Thread-132] result status is 255. com.huawei.bigdata.om.dbservice.backup.BackupTask.run(BackupTask.java:49)
2018-03-27 07:00:35,863 INFO  [Thread-132] result output is . com.huawei.bigdata.om.dbservice.backup.BackupTask.run(BackupTask.java:50)
2018-03-27 07:00:35,863 INFO  [Thread-132] result erro is ssh: connect to host 172.16.4.200 port 22: Connection refused
. com.huawei.bigdata.om.dbservice.backup.BackupTask.run(BackupTask.java:51)
2018-03-27 07:00:35,863 ERROR [Thread-132] DBService backup failed. com.huawei.bigdata.om.dbservice.backup.BackupTask.run(BackupTask.java:64)
2018-03-27 07:00:40,868 INFO  [pool-1-thread-5] query backup taskId is 6. com.huawei.bigdata.om.dbservice.backup.BackupRecoveryPlugin.getBackupProgress(BackupRecoveryPlugin.java:247)
```

Solution

Step 1 Log in to the active DBService node (the Master node bound with the DBService floating IP address).



Step 2 Add the DBService floating IP address to **ListenAddress** or comment out **ListenAddress** in the **/etc/ssh/sshd_config** file.

Step 3 Run the following command to restart the SSHD service:

service sshd restart

Step 4 Check whether the next DBServie backup is successful.

----End

16.5.9 DBService Failed to Start Due to the Loss of the DBService Configuration File

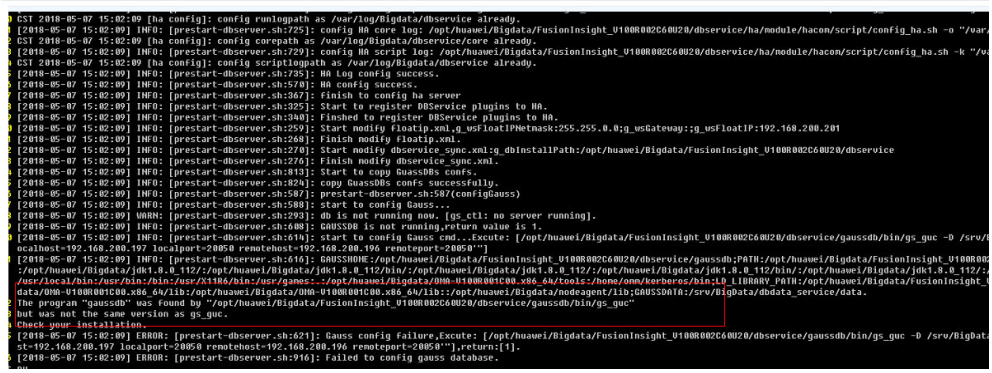
Symptom

The nodes are powered off unexpectedly, and the standby DBService node fails to be restarted.

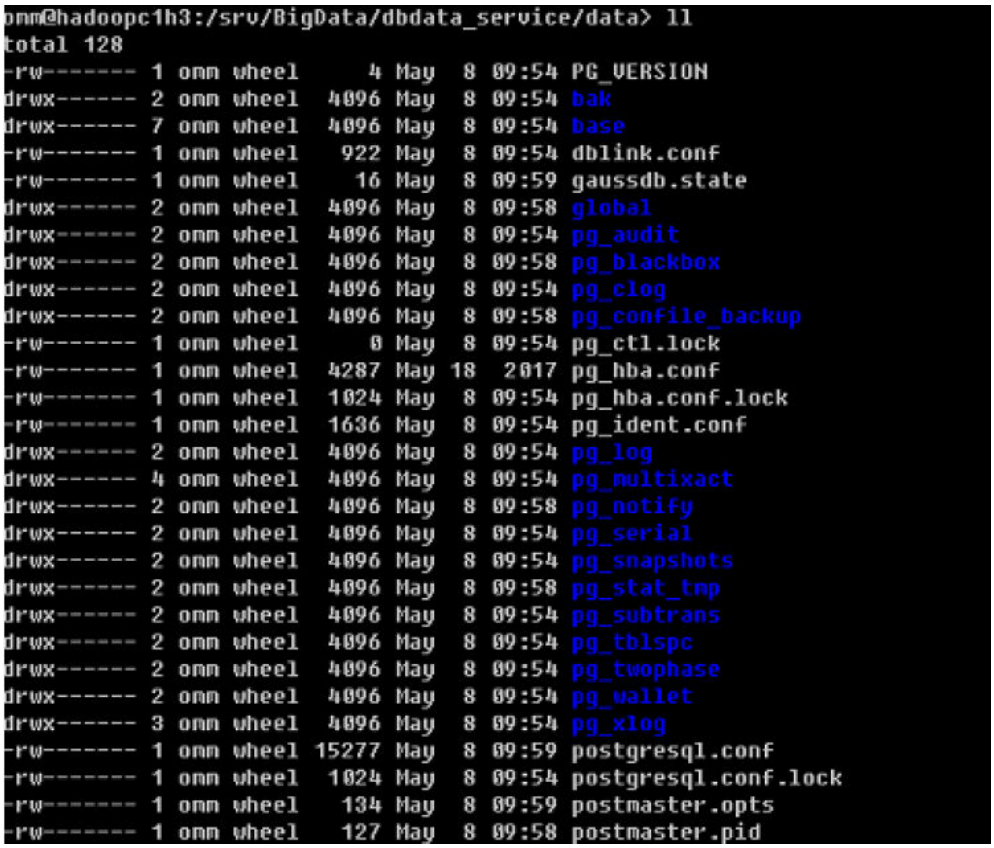
Cause Analysis

1. The **/var/log/Bigdata/dbservice/DB/gaussdb.log** file is viewed, which contains no information.
2. The **/var/log/Bigdata/dbservice/scriptlog/preStartDBService.log** file is viewed. This file contains the following information, indicating that the configuration information is lost:

```
The program "gaussdb" was found by "
/opt/Bigdata/MRS_xxx/install/dbservice/gaussdb/bin/gs_guc)
But not was not the same version as gs_guc.
Check your installation.
```



3. The configuration file in the **/srv/BigData/dbdata_service/data** directory on the active DBServer node is compared with the configuration file in the **/srv/BigData/dbdata_service/data** directory on the standby DBServer node, which shows major difference.



```
nm@hadoopc1h3:/srv/BigData/dbdata_service> cd data_bak/
nm@hadoopc1h3:/srv/BigData/dbdata_service/data_bak> ll
otal 64
rw------- 1 onm wheel   202 Feb 11 10:43 backup_label
rw------- 1 onm wheel     0 Feb 11 10:42 build_completed.start
rw------- 1 onm wheel    16 Apr 28 17:32 gaussdb.state
rw------- 1 onm wheel     7 Apr 28 17:32 gs_build.pid
rwx------ 2 onm wheel  4096 Feb 11 10:44 pg_audit
rwx------ 2 onm wheel  4096 Feb 11 10:41 pg_blackbox
rwx------ 2 onm wheel  4096 Feb 11 10:09 pg_confile_backup
rw------- 1 onm wheel     0 Apr 28 17:32 pg_ctl.lock
rw------- 1 onm wheel  4287 May 18  2017 pg_hba.conf
rwx------ 2 onm wheel  4096 Feb 11 10:43 pg_notify
rwx------ 2 onm wheel  4096 Feb 11 10:43 pg_xlog
rw------- 1 onm wheel 15155 May  7 15:33 postgresql.conf
rw------- 1 onm wheel  1024 May  7 15:33 postgresql.conf.lock
rw------- 1 onm wheel   134 Feb 11 10:42 postmaster.opts
```

Solution

Step 1 Copy the content in the **/srv/BigData/dbdata_service/data** directory on the active node to the standby node and ensure that the file permission and owner group are the same as those on the active node.

Step 2 Modify configuration in **postgresql.conf**. Set **localhost** to the IP of the local node and **remotehost** to the IP of the peer node.

```
#------------------------------------------------------------------------------
# CUSTOMIZED OPTIONS
#------------------------------------------------------------------------------

# Add settings for extensions here
max_files_per_process = 300
unix_socket_directory = '/var/run/FusionInsight-DBService'
replconninfo1 = 'localhost=192.168.200.197 localport=20050 remotehost=192.168.200.196 remoteport=20050'
"postgresql.conf" 382L, 15277C
```

Step 3 Log in to Manager and restart the standby DBServer node.

----End

16.6 Using Flink

16.6.1 "IllegalConfigurationException: Error while parsing YAML configuration file: "security.kerberos.login.keytab" Is Displayed When a Command Is Executed on an Installed Client

Symptom

After the client is successfully installed, an error message "IllegalConfigurationException: Error while parsing YAML configuration file:"security.kerberos.login.keytab" is displayed when the command (for example, **yarn-session.sh**) on the client is executed.

```
[root@8-5-131-10 bin]# yarn-session.sh
2018-10-25 01:22:06,454 | ERROR | [main] | Error while trying to split key and value in configuration file /opt/flinkclient/Flink/flink/conf/flink-conf.yaml:80: "security.kerberos.login.keytab: " | org.apache.flink.configuration.GlobalConfiguration (GlobalConfiguration.java:160)
Exception in thread "main" org.apache.flink.configuration.IllegalConfigurationException: Error while parsing YAML configuration file :80: "security.kerberos.login.keytab: "
```

```
        at org.apache.flink.configuration.GlobalConfiguration.loadYAMLResource(GlobalConfiguration.java:161)
        at org.apache.flink.configuration.GlobalConfiguration.loadConfiguration(GlobalConfiguration.java:112)
        at org.apache.flink.configuration.GlobalConfiguration.loadConfiguration(GlobalConfiguration.java:79)
        at org.apache.flink.yarn.cli.FlinkYarnSessionCli.main(FlinkYarnSessionCli.java:482)
[root@8-5-131-10 bin]#
```

Cause Analysis

In a secure cluster environment, Flink requires security authentication. The security authentication is not configured on the current client.

1. The following two authentication modes are available for Flink.
   - Kerberos authentication: Flink Yarn client, Yarn ResourceManager, JobManager, HDFS, TaskManager, Kafka, and ZooKeeper
   - Internal authentication mechanism of Yarn: The internal authentication used between YarnResource Manager and Application Master (AM).
2. If a security cluster is required, the Kerberos authentication and security cookie authentication are mandatory. As shown in the logs, it is found that the **security.kerberos.login.keytab** setting in the configuration file is incorrect and the security configuration is not performed.

Solution

Step 1 Download the keytab file from MRS and save it in a folder on a host where the Flink client resides.

Step 2 Configure following parameters in the **flink-conf.yaml** file:

1. Keytab path
   ```
   security.kerberos.login.keytab: /home/flinkuser/keytab/abc222.keytab
   ```

   NOTE

   - **/home/flinkuser/keytab/abc222.keytab** indicates the user directory, which is the directory saves the keytab file in **Step 1**.
   - Ensure that the client user has the permission on the corresponding directory.

2. Principal name
   ```
   security.kerberos.login.principal: abc222
   ```
3. In HA mode, if Zookeeper is configured, the ZooKeeper Kerberos authentication configuration items must be configured as follows:
   ```
   zookeeper.sasl.disable: false
   security.kerberos.login.contexts: Client
   ```
4. If Kerberos authentication is required between the Kafka client and Kafka broker, configure it as follows:
   ```
   security.kerberos.login.contexts: Client,KafkaClient
   ```

----End

16.6.2 "IllegalConfigurationException: Error while parsing YAML configuration file" Is Displayed When a Command Is Executed After Configurations of the Installed Client Are Changed

Symptom

After the client is successfully installed, an error message "IllegalConfigurationException: Error while parsing YAML configuration file: 81: "security.kerberos.login.principal:pippo " is displayed when the command (for example, **yarn-session.sh**) on the client is executed.

```
[root@8-5-131-10 bin]# yarn-session.sh
2018-10-25 19:27:01,397 | ERROR | [main] | Error while trying to split key and value in configuration file /opt/flinkclient/Flink/flink/conf/flink-conf.yaml:81: "security.kerberos.login.principal:pippo " | org.apache.flink.configuration.GlobalConfiguration (GlobalConfiguration.java:160)
Exception in thread "main" org.apache.flink.configuration.IllegalConfigurationException: Error while parsing YAML configuration file :81: "security.kerberos.login.principal:pippo "
        at org.apache.flink.configuration.GlobalConfiguration.loadYAMLResource(GlobalConfiguration.java:161)
        at org.apache.flink.configuration.GlobalConfiguration.loadConfiguration(GlobalConfiguration.java:112)
        at org.apache.flink.configuration.GlobalConfiguration.loadConfiguration(GlobalConfiguration.java:79)
        at org.apache.flink.yarn.cli.FlinkYarnSessionCli.main(FlinkYarnSessionCli.java:482)
```

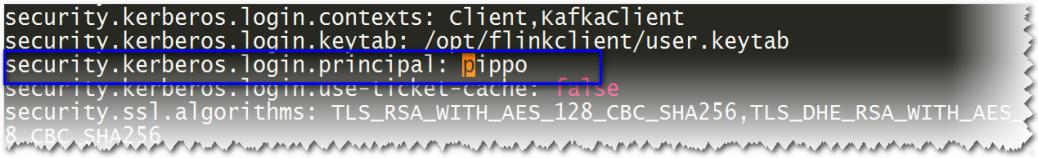
Cause Analysis

The **security.kerberos.login.principal:pippo** item in the **flink-conf.yaml** configuration file was faulty.

Solution

Modify the configuration in the **flink-conf.yaml** file.

Note: The configuration item name and value must be separated by a space.



16.6.3 The yarn-session.sh Command Fails to Be Executed When the Flink Cluster Is Created

Symptom

During the creation of the Flink cluster, an error message is displayed after the **yarn-session.sh** command execution is suspended.

```
2018-09-20 22:51:16,842 | WARN  | [main] | Unable to get ClusterClient status from Application Client | org.apache.flink.yarn.YarnClusterClient (YarnClusterClient.java:253)
```

```
org.apache.flink.util.FlinkException: Could not connect to the leading JobManager. Please check that the JobManager is running.
    at org.apache.flink.client.program.ClusterClient.getJobManagerGateway(ClusterClient.java:861)
    at org.apache.flink.yarn.YarnClusterClient.getClusterStatus(YarnClusterClient.java:248)
    at org.apache.flink.yarn.YarnClusterClient.waitForClusterToBeReady(YarnClusterClient.java:516)
    at org.apache.flink.yarn.cli.FlinkYarnSessionCli.run(FlinkYarnSessionCli.java:717)
    at org.apache.flink.yarn.cli.FlinkYarnSessionCli$1.call(FlinkYarnSessionCli.java:514)
    at org.apache.flink.yarn.cli.FlinkYarnSessionCli$1.call(FlinkYarnSessionCli.java:511)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.Subject.doAs(Subject.java:422)
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1729)
    at org.apache.flink.runtime.security.HadoopSecurityContext.runSecured(HadoopSecurityContext.java:41)
    at org.apache.flink.yarn.cli.FlinkYarnSessionCli.main(FlinkYarnSessionCli.java:511)
Caused by: org.apache.flink.runtime.leaderretrieval.LeaderRetrievalException: Could not retrieve the leader gateway.
    at org.apache.flink.runtime.util.LeaderRetrievalUtils.retrieveLeaderGateway(LeaderRetrievalUtils.java:79)
    at org.apache.flink.client.program.ClusterClient.getJobManagerGateway(ClusterClient.java:856)
    ... 10 common frames omitted
Caused by: java.util.concurrent.TimeoutException: Futures timed out after [10000 milliseconds]
```

Possible Causes

The SSL communication encryption is enabled for Flink, but no correct SSL certificate is configured.

Solution

For MRS 2.*x* or earlier, perform the following operations:

Method 1:

Run the following command to disable the Flink SSL communication encryption, and modify the client configuration file **conf/flink-conf.yaml**.

```
security.ssl.internal.enabled: false
```

Method 2:

Enable the Flink SSL communication encryption and retain the default value of **security.ssl.internal.enabled**. Configure the SSL as follows:

- If the KeyStore or TrustStore file is a relative path, and the Flink client directory where the command is executed can directly access this relative path.

  ```
  security.ssl.internal.keystore: ssl/flink.keystore
  security.ssl.internal.truststore: ssl/flink.truststore
  ```

  Add **-t** option to the CLI **yarn-session.sh** command of Flink to transmit the KeyStore and TrustStore files to each execution node. Example:

  ```
  yarn-session.sh -t ssl/ 2
  ```

- If the keystore or truststore file path is an absolute path, the keystore or truststore files must exist in the absolute path on Flink Client and all nodes.

  ```
  security.ssl.internal.keystore: /opt/client/Flink/flink/conf/flink.keystore
  security.ssl.internal.truststore: /opt/client/Flink/flink/conf/flink.truststore
  ```

For MRS 3.*x* or later, perform the following operations:

Method 1:

Run the following command to disable the Flink SSL communication encryption, and modify the client configuration file **conf/flink-conf.yaml**.

```
security.ssl.enabled: false
```

Method 2:

Enable the Flink SSL communication encryption and retain the default value of **security.ssl.enabled**. Configure the SSL as follows:

- If the KeyStore or TrustStore file is a relative path, and the Flink client directory where the command is executed can directly access this relative path.

  ```
  security.ssl.keystore: ssl/flink.keystore
  security.ssl.truststore: ssl/flink.truststore
  ```

  Add **-t** option to the CLI **yarn-session.sh** command of Flink to transmit the KeyStore and TrustStore files to each execution node. Example:

  ```
  yarn-session.sh -t ssl/ 2
  ```

- If the keystore or truststore file path is an absolute path, the keystore or truststore files must exist in the absolute path on Flink Client and all nodes.

  ```
  security.ssl.keystore: /opt/Bigdata/client/Flink/flink/conf/flink.keystore
  security.ssl.truststore: /opt/Bigdata/client/Flink/flink/conf/flink.truststore
  ```

16.6.4 Failed to Create a Cluster by Executing the yarn-session Command When a Different User Is Used

Symptom

Two users **testuser** and **bdpuser** with the same rights are used to create the Flink cluster.

When user **testuser** is used to create a Flink cluster, no error message is displayed. While user **bdpuser** is used to create a Flink cluster, an error message is displayed during the **yarn-session.sh** command execution:

```
2019-01-02 14:28:09,098 | ERROR | [main] | Ensure path threw exception |
org.apache.flink.shaded.curator.org.apache.curator.framework.imps.CuratorFrameworkImpl
(CuratorFrameworkImpl.java:566)
org.apache.flink.shaded.zookeeper.org.apache.zookeeper.KeeperException$NoAuthException:
KeeperErrorCode = NoAuth for /flink/application_1545397824912_0022
```

Possible Causes

The HA configuration item is not modified. In the Flink configuration file, the default value of **high-availability.zookeeper.client.acl** is **creator**, indicating that only the creator has the access permission. A new user cannot access the directory on ZooKeeper. As a result, the **yarn-session.sh** command execution fails.

Solution

Step 1 Modify the value of **high-availability.zookeeper.path.root** in the **conf/flink-conf.yaml** file. For example, run the following command:

```
high-availability.zookeeper.path.root: flink2
```

Step 2 Submit the tasks again.

----End

16.6.5 Flink Service Program Fails to Read Files on the NFS Disk

Issue

The Flink service program cannot read files on the NFS disk mounted to the cluster node.

Symptom

The Flink service program developed by a user needs to read the user-defined configuration file. The configuration file is stored on the NFS disk. The NFS disk is mounted to the cluster node and can be accessed by all nodes in the cluster. After the user submits the Flink program, the service code cannot access the user-defined configuration file. As a result, the service program fails to be started.

Cause Analysis

The root cause is that the permission on the root directory of the NFS disk is insufficient. As a result, the Flink program cannot access the directory after being started.

Flink tasks of MRS are running on Yarn. If the cluster is a common cluster, the user who runs the tasks on Yarn is **yarn_user**. If the user-defined configuration file is used after the tasks are started, **yarn_user** must be allowed to access the file and the parent directory of the file (parent directory of the file on the NFS, not the soft link on the cluster node). Otherwise, the program cannot obtain the file content. If the cluster is a cluster with Kerberos authentication enabled, the file permission must allow the user who submits the program to access the file.

Procedure

Step 1 Log in to the Master node in the cluster as user **root**.

Step 2 Run the following command to check the permission on the parent directory of the user-defined configuration file:

ll <*Parent directory of the file path***>**

Step 3 Go to the directory of the file to be accessed on the NFS disk and change the permission of the parent directory of the user-defined configuration file to 755.

chmod 755 -R /<*Path of the parent directory of the file***>**

Step 4 Check whether the Core or Task node can access the configuration file.

1. Log in to the Core or Task node as the **root** user.

   If Kerberos authentication is enabled for the current cluster, log in to the Core node as user **root**.

2. Run **su - yarn_user** to switch to user **yarn_user**.

   If Kerberos authentication is enabled for the cluster, run the **su -** *User who submits the job* command to switch the user.

3. Run the following command to check the user permission. The file path must be the absolute path of the file.

ll <*File path*>

----End

Summary and Suggestions

When a user-defined configuration file needs to be accessed in the submitted task, especially when the NFS disk is mounted, you need to check whether the permission of the parent directory of the file is correct in addition to the file permission. When an NFS disk is mounted to an MRS cluster node, a soft link is created to the NFS directory. In this case, you need to check whether the directory permission on the NFS is correct.

16.6.6 Failed to Customize the Flink Log4j Log Level

Issue

The customized level for Flink Log4j logs of an MRS 3.1.0 cluster does not take effect.

Symptom

1. When analyzing data using Flink of an MRS 3.1.0 cluster, a user changes the log level in the **log4j.properties** file in the **$Flink_HOME/conf** directory to **INFO**.
2. However, after the task is submitted successfully, the log level displayed on the console is still **ERROR**, rather than **INFO**.

Cause Analysis

The **log4j.properties** file in the **$Flink_HOME/conf** directory controls the log output of in JobManager and TaskManager operators, and the logs are printed to the corresponding Yarn containers. You can view the logs on the Yarn web UI. In MRS 3.1.0 and later versions, the default log framework of Flink 1.12.0 is Log4j2. The configuration method is different from that of Log4j. For example, Log4j log rules do not take effect.

Procedure

For details about configuring Log4j2 log specifications, see the official open-source document at **http://logging.apache.org/log4j/2.x/manual/configuration.html#Properties**.

16.7 Using Flume

16.7.1 Class Cannot Be Found After Flume Submits Jobs to Spark Streaming

Issue

After Flume submits jobs to Spark Streaming, the class cannot be found.

Symptom

After the Spark Streaming code is packed into a JAR file and submitted to the cluster, an error message is displayed indicating that the class cannot be found. The following two methods are not useful:

1. When submitting a Spark job, run the **--jars** command to reference the JAR file of the class.
2. Import the JAR file where the class resides to the JAR file of Spark Streaming.

Cause Analysis

Some JAR files cannot be loaded during Spark job execution, resulting that the class cannot be found.

Procedure

Step 1 Run the **--jars** command to load the **flume-ng-sdk-{version} .jar** dependency package.

Step 2 Modify the two configuration items in the **spark-default.conf** file:

spark.driver.extraClassPath=$PWD/*: {Add the original value}

spark.executor.extraClassPath =$PWD/*

Step 3 Run the job successfully. If an error is reported, check which JAR is not loaded and perform step 1 and step 2 again.

----End

16.7.2 Failed to Install a Flume Client

Symptom

A Flume client fails to be installed, and "JAVA_HOME is null" or "flume has been installed" is displayed.

```
CST 2016-08-31 17:02:51 [flume-client install]: JAVA_HOME is null in current user,please install the JDK and set the JAVA_HOME
CST 2016-08-31 17:02:51 [flume-client install]: check environment failed.
CST 2016-08-31 17:02:51 [flume-client install]: check param failed.
CST 2016-08-31 17:02:51 [flume-client install]: install flume client failed.
```

```
CST 2016-08-31 17:03:58 [flume-client install]: flume has been installed
CST 2016-08-31 17:03:58 [flume-client install]: check path failed.
CST 2016-08-31 17:03:58 [flume-client install]: check param failed.
CST 2016-08-31 17:03:58 [flume-client install]: install flume client failed.
```

Cause Analysis

- Environment variables are checked during Flume client installation. If no Java is available, an error message is displayed stating "JAVA_HOME is null" and the installation quits.
- If Flume has been installed in the specified directory, an error message is displayed stating "flume has been installed" during client installation and the installation quits.

Solution

Step 1 Run the following command if an error message is displayed stating "JAVA_HOME is null":

export JAVA_HOME=*Java path*

Set **JAVA_HOME** and execute the installation script again.

Step 2 If a Flume client has been installed under the specified directory, uninstall the client and use another directory.

----End

16.7.3 A Flume Client Cannot Connect to the Server

Symptom

A user installs a Flume client and sets an Avro sink to communicate with the server. However, the Flume server cannot be connected.

Cause Analysis

1. The server is incorrectly configured and the monitoring port fails to be started up. For example, an incorrect IP address or an occupied port is configured for the Avro source of the server. View Flume run logs.

```
2016-08-31 17:28:42,092 | ERROR | [lifecycleSupervisor-1-9] |  Unable to start EventDrivenSourceRunner: { source:Avro source avro_source: { bindAddress: 10.120.205.7, port: 21154 } } - Exception follows.  | org.apache.flume.lifecycle.LifecycleSupervisor $MonitorRunnable.run(LifecycleSupervisor.java:253)
java.lang.RuntimeException: org.jboss.netty.channel.ChannelException: Failed to bind to: /192.168.205.7:21154
```

2. If encrypted transmission is used, the certificate or password is incorrect.

```
2016-08-31 17:15:59,593 | ERROR | [conf-file-poller-0] |  Source avro_source has been removed due to an error during configuration  | org.apache.flume.node.AbstractConfigurationProvider.loadSources(AbstractConfigurationProvider.java:388)
org.apache.flume.FlumeException: Avro source configured with invalid keystore: /opt/Bigdata/MRS_XXX/install/FusionInsight-Flume-1.9.0/flume/conf/flume_sChat.jks
```

3. The network connection between the client and the server is abnormal.

```
PING 192.168.85.55 (10.120.85.55) 56(84) bytes of data.
From 192.168.85.50 icmp_seq=1 Destination Host Unreachable
From 192.168.85.50 icmp_seq=2 Destination Host Unreachable
From 192.168.85.50 icmp_seq=3 Destination Host Unreachable
From 192.168.85.50 icmp_seq=4 Destination Host Unreachable
```

Solution

Step 1 Set a correct IP address (an IP address of the local host). If the port has been occupied, configure another free port.

Step 2 Configure a correct certificate path.

Step 3 Contact the network administrator to restore the network.

----End

16.7.4 Flume Data Fails to Be Written to the Component

Symptom

After the Flume process is started, Flume data cannot be written to the corresponding component. (The following uses writing data from the server to HDFS as an example.)

Cause Analysis

1. HDFS is not started or is faulty. View Flume run logs.

```
2019-02-26 11:16:33,564 | ERROR | [SinkRunner-PollingRunner-DefaultSinkProcessor] |  opreation the hdfs file errors.  | org.apache.flume.sink.hdfs.HDFSEventSink.process(HDFSEventSink.java:414)
2019-02-26 11:16:33,747 | WARN  | [hdfs-CCCC-call-runner-4] |  A failover has occurred since the start of call #32795 ClientNamenodeProtocolTranslatorPB.getFileInfo over 192-168-13-88/192.168.13.88:25000  | org.apache.hadoop.io.retry.RetryInvocationHandler$ProxyDescriptor.failover(RetryInvocationHandler.java:220)
2019-02-26 11:16:33,748 | ERROR | [hdfs-CCCC-call-runner-4] |  execute hdfs error. {}  | org.apache.flume.sink.hdfs.HDFSEventSink$3.call(HDFSEventSink.java:744)
java.net.ConnectException: Call From 192-168-12-221/192.168.12.221 to 192-168-13-88:25000 failed on connection exception: java.net.ConnectException: Connection refused; For more details see:  http://wiki.apache.org/hadoop/ConnectionRefused
```

2. The HDFS sink is not started. Check the Flume run log. It is found that the Flume current metrics file does not contain sink information.

```
2019-02-26 11:46:05,501 | INFO  | [pool-22-thread-1] |  flume current metrics:{"CHANNEL.BBBB":{"ChannelCapacity":"10000","ChannelFillPercentage":"0.0","Type":"CHANNEL","ChannelStoreSize":"0","EventProcessTimedelta":"0","EventTakeSuccessCount":"0","ChannelSize":"0","EventTakeAttemptCount":"0","StartTime":"1551152734999","EventPutAttemptCount":"0","EventPutSuccessCount":"0","StopTime":"0"},"SOURCE.AAAA":{"AppendBatchAcceptedCount":"0","EventAcceptedCount":"0","AppendReceivedCount":"0","MonTime":"0","StartTime":"1551152735503","AppendBatchReceivedCount":"0","EventReceivedCount":"0","Type":"SOURCE","TotalFilesCount":"1001","SizeAcceptedCount":"0","UpdateTime":"605410241202740","AppendAcceptedCount":"0","OpenConnectionCount":"0","MovedFilesCount":"1001","StopTime":"0"}}  | org.apache.flume.node.Application.getRestartComps(Application.java:467)
```

Solution

Step 1 If the component to which Flume writes data is not started, start the component. If the component is abnormal, contact technical support.

Step 2 If the sink is not started, check whether the configuration file is correctly configured. If the configuration file is incorrectly configured, modify the configuration file and restart the Flume process. If the configuration file is correctly configured, view the error information in the log and rectify the fault based on the error information.

----End

16.7.5 Flume Server Process Fault

Symptom

After Flume runs for a period of time, the Flume instance is in the faulty state on Manager.

Cause Analysis

If the Flume file or folder permission is abnormal, the following information is displayed on MRS Manager after the restart:

```
[2019-02-26 13:38:02]RoleInstance prepare to start failure [{ScriptExecutionResult=ScriptExecutionResult [exitCode=126, output=, errMsg=sh: line 1: /opt/Bigdata/MRS_XXX/install/FusionInsight-Flume-1.9.0/flume/bin/flume-manage.sh: Permission denied
```

Solution

Compare the file and folder permissions with those for the Flume node that is running properly and correct the file or folder permissions.

16.7.6 Flume Data Collection Is Slow

Symptom

After Flume is started, it takes a long time for Flume to collect data.

Cause Analysis

1. The heap memory of Flume is not properly set. As a result, the Flume process keeps in the GC state. View Flume run logs.
   ```
   2019-02-26T13:06:20.666+0800: 1085673.512: [Full GC:[CMS: 3849339k->3843458K(3853568K), 2.5817610 secs] 4153654K->3843458K(4160256K), [CMS Perm : 27335K->27335K(45592K),2.5820080 SECS] [Times: user=2.63, sys0.00, real=2.59 secs]
   ```
2. The **deletePolicy** policy configured for the Spooldir source is **immediate**.

Solution

Step 1 Increase the size of the heap memory (**xmx**).

Step 2 Change the **deletePolicy** policy of the Spooldir source to **never**.

----End

16.7.7 Failed to Start Flume

Symptom

The Flume service fails to be installed or restarted.

Cause Analysis

1. The heap memory of Flume is greater than the remaining memory of the server. The Flume startup log shows the following information:

```
[CST 2019-02-26 13:31:43][INFO] [[checkMemoryValidity:124]] [GC_OPTS is invalid:
Xmx(40960000MB) is bigger than the free memory(56118MB) in system.] [9928]
```

2. The permission on the Flume file or folder is abnormal. The following information is displayed on the GUI or in the background:

```
[2019-02-26 13:38:02]RoleInstance prepare to start failure
[{ScriptExecutionResult=ScriptExecutionResult [exitCode=126, output=, errMsg=sh: line 1: /opt/Bigdata/
MRS_XXX/install/FusionInsight-Flume-1.9.0/flume/bin/flume-manage.sh: Permission denied
```

3. The **JAVA_HOME** is incorrectly configured. The Flume agent startup log shows the following information:

```
Info: Sourcing environment configuration script /opt/FlumeClient/fusioninsight-flume-1.9.0/conf/
flume-env.sh
+ '[' -n '' ']'
+ exec /tmp/MRS-Client/MRS_Flume_ClientConfig/JDK/jdk-8u18/bin/java '-
XX:OnOutOfMemoryError=bash /opt/FlumeClient/fusioninsight-flume-1.9.0/bin/
out_memory_error.sh /opt/FlumeClient/fusioninsight-flume-1.9.0/conf %p' -Xms2G -Xmx4G -
XX:CMSFullGCsBeforeCompaction=1 -XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -
XX:+UseCMSCompactAtFullCollection -Dkerberos.domain.name=hadoop.hadoop.com -verbose:gc -
XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M -XX:+PrintGCDetails -
XX:+PrintGCDateStamps -Xloggc:/var/log/Bigdata//flume-client-1/flume/flume-root-20190226134231-
%p-gc.log -Dproc_org.apache.flume.node.Application -Dproc_name=client -Dproc_conf_file=/opt/
FlumeClient/fusioninsight-flume-1.9.0/conf/properties.properties -Djava.security.krb5.conf=/opt/
FlumeClient/fusioninsight-flume-1.9.0/conf//krb5.conf -Djava.security.auth.login.config=/opt/
FlumeClient/fusioninsight-flume-1.9.0/conf//jaas.conf -Dzookeeper.server.principal=zookeeper/
hadoop.hadoop.com -Dzookeeper.request.timeout=120000 -Dflume.instance.id=884174180 -
Dflume.agent.name=clientName1 -Dflume.role=client -Dlog4j.configuration.watch=true -
Dlog4j.configuration=log4j.properties -Dflume_log_dir=/var/log/Bigdata//flume-client-1/flume/ -
Dflume.service.id=flume-client-1 -Dbeetle.application.home.path=/opt/FlumeClient/fusioninsight-
flume-1.9.0/conf/service -Dflume.called.from.service -Dflume.conf.dir=/opt/FlumeClient/fusioninsight-
flume-1.9.0/conf -Dflume.metric.conf.dir=/opt/FlumeClient/fusioninsight-flume-1.9.0/conf -
Dflume.script.home=/opt/FlumeClient/fusioninsight-flume-1.9.0/bin -cp '/opt/FlumeClient/
fusioninsight-flume-1.9.0/conf:/opt/FlumeClient/fusioninsight-flume-1.9.0/lib/*:/opt/FlumeClient/
fusioninsight-flume-1.9.0/conf/service/' -Djava.library.path=/opt/FlumeClient/fusioninsight-flume-1.9.0/
plugins.d/native/native org.apache.flume.node.Application --conf-file /opt/FlumeClient/fusioninsight-
flume-1.9.0/conf/properties.properties --name client
/opt/FlumeClient/fusioninsight-flume-1.9.0/bin/flume-ng: line 233: /tmp/FusionInsight-Client/Flume/
FusionInsight_Flume_ClientConfig/JDK/jdk-8u18/bin/java: No such file or directory
```

Solution

Step 1 Increase the size of the heap memory (**xmx**).

Step 2 Compare the file and folder permissions with those for node where Flume is started properly and change the incorrect file or folder permissions.

Step 3 Reconfigure **JAVA_HOME**. On the client, replace the value of **JAVA_HOME** in the **${install_home}/fusioninsight-flume-***Flume version***/conf/ENV_VARS** file. On the server, replace the value of **JAVA_HOME** in the **ENV_VARS** file in the **etc** directory.

To obtain the value of **JAVA_HOME**, log in to the node where Flume is properly started and run the **echo ${JAVA_HOME}** command.

NOTE

${install_home} is the installation path of the Flume client.

----End

16.8 Using HBase

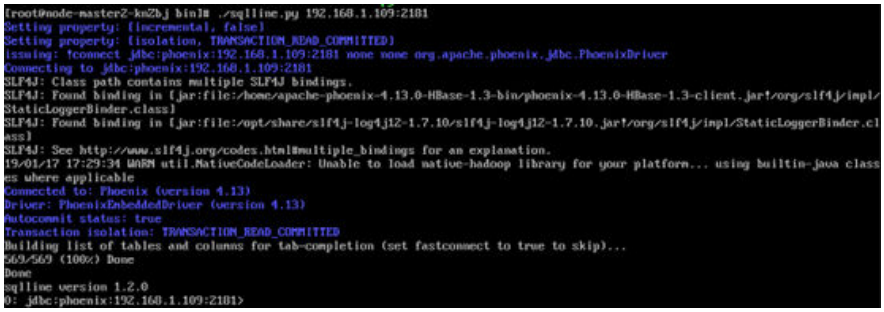
16.8.1 Slow Response to HBase Connection

Issue

Under the same VPC network, response is slow when an external cluster connects to HBase through Phoenix.

Symptom

Under the same VPC network, response is slow when an external cluster connects to HBase through Phoenix.



Possible Cause

DNS has been configured. When a client connects to HBase, DNS resolves the server first, causing slow response.

Procedure

Step 1 Log in to the Master node as user **root**.

Step 2 Run the **vi /etc/resolv.conf** command to open the **resolv.conf** file and comment out the address of the DNS server, for example, #1.1.1.1.

----End

16.8.2 Failed to Authenticate the HBase User

Issue

Failed to authenticate the HBase user.

Symptom

Failed to authenticate the HBase user on the client. The following error information is displayed:

```
2019-05-13 10:53:09,975 ERROR [localhost-startStop-1] xxxConfig.LoginUtil: login failed with hbaseuser and /usr/local/linoseyc/hbase-tomcat/webapps/bigdata_hbase/WEB-INF/classes/user.keytab.
2019-05-13 10:53:09,975 ERROR [localhost-startStop-1] xxxConfig.LoginUtil: perhaps cause 1 is (wrong password) keytab file and user not match, you can kinit -k -t keytab user in client server to check.
2019-05-13 10:53:09,975 ERROR [localhost-startStop-1] xxxConfig.LoginUtil: perhaps cause 2 is (clock skew) time of local server and remote server not match, please check ntp to remote server.
2019-05-13 10:53:09,975 ERROR [localhost-startStop-1] xxxConfig.LoginUtil: perhaps cause 3 is (aes256 not support) aes256 not support by default jdk/jre, need copy local_policy.jar and US_export_policy.jar from remote server in path ${BIGDATA_HOME}/jdk/jre/lib/security.
```

Cause Analysis

The version of the JAR file in the JDK used by the user is different from that of the JAR file authenticated by MRS.

Procedure

Step 1 Log in to the Master1 node as user **root**.

Step 2 Run the following command to check the JAR file authenticated by MRS:

ll /opt/share/local_policy/local_policy.jar

ll /opt/Bigdata/jdk{version}/jre/lib/security/local_policy.jar

Step 3 Download the JAR package queried in step 2 to the local host.

Step 4 Copy the downloaded JAR package to the local JDK directory **/opt/Bigdata/jdk/jre/lib/security**.

Step 5 Run the **cd /opt/client/HBase/hbase/bin** command to go to the **bin** directory of HBase.

Step 6 Run the **sh start-hbase.sh** command to restart HBase.

----End

16.8.3 RegionServer Failed to Start Because the Port Is Occupied

Symptom

RegionServer is in the **Restoring** state on Manager.

Cause Analysis

1. View the RegionServer log (**/var/log/Bigdata/hbase/rs/hbase-omm-xxx.log**).
2. Run the **lsof -i:21302** command (the port number of MRS 1.7.X and later versions is 16020) to view the PID. Based on the PID, check the process. It is found that the RegionServer port is occupied by DFSZkFailoverController.
3. The value of **/proc/sys/net/ipv4/ip_local_port_range** is **9000 65500**. The temporary port range and the MRS port range overlap. This is because the preinstall operation is not performed during installation.

Solution

Step 1 Run the **kill -9** *DFSZkFailoverController pid* command to ensure that another port is bound with after a restart and restart the RegionServer in the **Restoring** state.

----End

16.8.4 HBase Failed to Start Due to Insufficient Node Memory

Symptom

The RegionServer service of HBase is always in the **Restoring** state.

Cause Analysis

1. Check the RegionServer log (**/var/log/Bigdata/hbase/rs/hbase-omm-XXX.out**). It is found that the following information is printed:
   ```
   There is insufficient memory for the Java Runtime Environment to continue.
   ```
2. Run the **free** command to check the memory. It is found that the available memory of the node is insufficient.

Solution

Step 1 Locate why the memory is insufficient. It is found that some processes occupy too much memory or the server does not have sufficient memory.

----End

16.8.5 HBase Service Unavailable Due to Poor HDFS Performance

Symptom

The HBase component intermittently reports alarms indicating that the service is unavailable.

Cause Analysis

HDFS performance is low, causing health check timeout and the alarm is generated accordingly. You can perform the following operations:

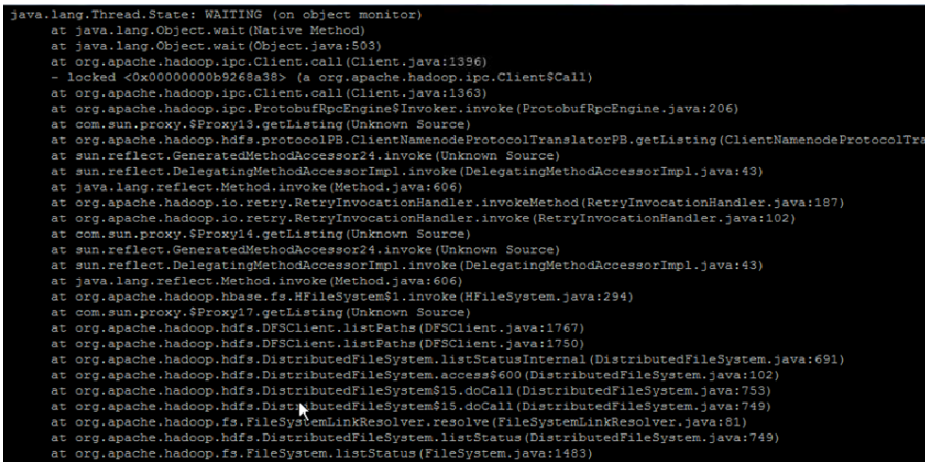
1. View the HMaster log (**/var/log/Bigdata/hbase/hm/hbase-omm-xxx.log**) and check that **system pause**, **jvm**, and other GC-related information is not frequently printed in the log.
2. Determine whether the fault is caused by poor HDFS performance using either of the following methods:
   a. Run **hbase shell** to access the HBase shell, and run the **list** command to check whether it takes a long period of time to list all tables in HBase.
   b. Enable printing of the debug logs of HDFS, and check whether it takes a long period of time to list the content of a large number of directories by running the **hadoop fs –ls** */XXX/XXX* command.
   c. Print the Java stack information about a specified HMaster process.

      su - omm

      jps

      jstack pid
3. Check the jstack information. The following figure shows that the process is stuck at the **DFSClient.listPaths** state.

Figure 16-20 Exception



Solution

Step 1 If this alarm is caused by poor HDFS performance, check whether Impala is of an earlier version or JournalNode was incorrectly deployed during the initial deployment (more than three JournalNode nodes are deployed).

----End

16.8.6 HBase Failed to Start Due to Inappropriate Parameter Settings

Symptom

After some parameters are modified, HBase cannot be started.

Cause Analysis

1. Check the HMaster log (**/var/log/Bigdata/hbase/hm/hbase-omm-*xxx*.log**). It is found that the total of **hbase.regionserver.global.memstore.size** and **hfile.block.cache.size** is greater than 0.8, which causes the startup failure. Therefore, adjust the parameter values to make sure that the total value is less than 0.8.

2. Check the HMaster and RegionServer out logs (**/var/log/Bigdata/hbase/hm/hbase-omm-*xxx*.out/var/log/Bigdata/hbase/rs/hbase-omm-*xxx*.out**). It is found that **Unrecognized VM option** is displayed.

```
Unrecognized VM option
Error: Could not create the Java Virtual Machine.
Error: A fatal exception has occurred. Program will exit.
```

Check the **GC_OPTS** parameters. It is found that the parameters contain unnecessary spaces, for example, **-D sun.rmi.dgc.server.gcInterval=0x7FFFFFFFFFFFFE**.

Solution

Step 1 After the **MemStore** and **cache** parameters are modified, the HBase service is restarted successfully.

Step 2 After the **GC_OPTS** parameters are modified, the HBase service is restarted successfully.

----End

16.8.7 RegionServer Failed to Start Due to Residual Processes

Symptom

The HBase service fails to start, and an error is reported during the health check.

Cause Analysis

Check detailed information about HBase startup on the MRS Manager page. It is found that **the previous process is not quit** is displayed.

Solution

Step 1 Log in to the node and run the **ps -ef | grep HRegionServer** command in the background. A residual process is found.

Step 2 After confirming that the process can be killed, kill the process. If the process cannot be stopped by running the **kill** command, run the **kill -9** command to forcibly stop the process.

Step 3 Restart the HBase service.

----End

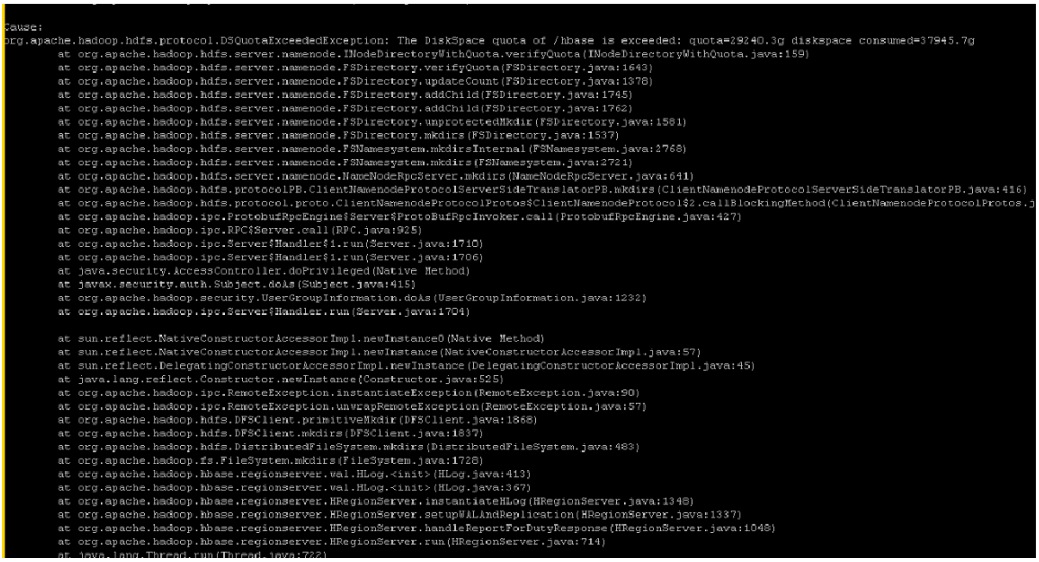
16.8.8 HBase Failed to Start Due to a Quota Set on HDFS

Symptom

HBase fails to start.

Cause Analysis

Check the HMaster log (**/var/log/Bigdata/hbase/hm/hbase-omm-*xxx*.log**). It is found that "The DiskSpace quota of /hbase is exceeded" is displayed.



Solution

Step 1 Run the **df -h** command to check data directory space. It is found that the directory space is full. Delete unnecessary data to free up space.

Step 2 Expand the node to ensure that the data directory space is sufficient.

----End

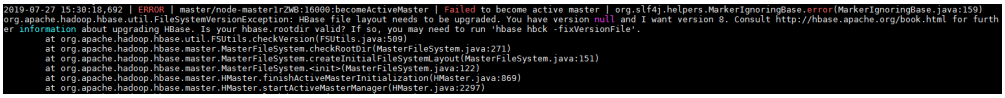
16.8.9 HBase Failed to Start Due to Corrupted Version Files

Symptom

HBase fails to start.

Cause Analysis

1. The **hbase.version** file is read during HBase startup. However, the log indicates that a reading exception occurs.



2. The file cannot be viewed by running the **hadoop fs -cat /hbase/hbase.version** command. The file is corrupted.

Solution

Step 1 Run the **hbase hbck -fixVersionFile** command to restore the file.

Step 2 If the problem persists after performing **Step 1**, obtain the **hbase.version** file from another cluster of the same version and upload the file to replace the original one.

Step 3 Restart the HBase service.

----End

16.8.10 High CPU Usage Caused by Zero-Loaded RegionServer

Symptom

The CPU usage of RegionServer is high, but there is no service running on RegionServer.

Cause Analysis

1. Run the **top** command to obtain the CPU usage of RegionServer processes and check the IDs of processes with high CPU usage.
2. Obtain the CPU usage of threads under these processes based on the RegionServer process IDs.

   Run the **top -H -p** *<PID>* (replace it with the actual RegionServer process ID). As shown in the following figure, the CPU usage of some threads reaches 80%.

   ```
   PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
   75706 omm     20   0 6879444   1.0g  25612 S  90.4  1.6   0:00.00 java
   75716 omm     20   0 6879444   1.0g  25612 S  90.4  1.6   0:04.74 java
   75720 omm     20   0 6879444   1.0g  25612 S  88.6  1.6   0:01.93 java
   75721 omm     20   0 6879444   1.0g  25612 S  86.8  1.6   0:01.99 java
   75722 omm     20   0 6879444   1.0g  25612 S  86.8  1.6   0:01.94 java
   75723 omm     20   0 6879444   1.0g  25612 S  86.8  1.6   0:01.96 java
   75724 omm     20   0 6879444   1.0g  25612 S  86.8  1.6   0:01.97 java
   75725 omm     20   0 6879444   1.0g  25612 S  81.5  1.6   0:02.06 java
   75726 omm     20   0 6879444   1.0g  25612 S  79.7  1.6   0:02.01 java
   75727 omm     20   0 6879444   1.0g  25612 S  79.7  1.6   0:01.95 java
   75728 omm     20   0 6879444   1.0g  25612 S  78.0  1.6   0:01.99 java
   ```

3. Obtain the thread stack information based on the ID of the RegionServer process.

   jstack 12345 >allstack.txt (Replace it with the actual RegionServer process ID.)

4. Convert the thread ID into the hexadecimal format:

   printf "%x\n" 30648

   In the command output, the TID is **77b8**.

5. Search the thread stack based on the hexadecimal TID. It is found that the compaction operation is performed.

   ```
   "regionserver/ahbd-hbase-dat1/12.2.168.1:21302-longCompactions-1482676601478" #1641 prio=5 os_prio=0 tid=0x00007fa614563000 nid=0x77b8 runnable [0x0
      java.lang.Thread.State: RUNNABLE
       at org.apache.hadoop.io.compress.snappy.SnappyCompressor.compressBytesDirect(Native Method)
       at org.apache.hadoop.io.compress.snappy.SnappyCompressor.compress(SnappyCompressor.java:228)
       at org.apache.hadoop.io.compress.BlockCompressorStream.compress(BlockCompressorStream.java:149)
       at org.apache.hadoop.io.compress.BlockCompressorStream.finish(BlockCompressorStream.java:142)
       at org.apache.hadoop.hbase.io.encoding.HFileBlockDefaultEncodingContext.compressAfterEncoding(HFileBlockDefaultEncodingContext.java:219)
       at org.apache.hadoop.hbase.io.encoding.HFileBlockDefaultEncodingContext.compressAndEncrypt(HFileBlockDefaultEncodingContext.java:132)
       at org.apache.hadoop.hbase.io.hfile.HFileBlock$Writer.finishBlock(HFileBlock.java:989)
       at org.apache.hadoop.hbase.io.hfile.HFileBlock$Writer.ensureBlockReady(HFileBlock.java:961)
       at org.apache.hadoop.hbase.io.hfile.HFileBlock$Writer.finishBlockAndWriteHeaderAndData(HFileBlock.java:1077)
   ```

6. Perform the same operations on other threads. It is found that the threads are compaction threads.

   ```
   "regionserver/ahbd-hbase-dat1/12.2.168.1:21302-longCompactions-1482676601473" #1629 prio=5 os_prio=0 tid=0x00007fa61454d800 nid=0x77a0 runnable
      java.lang.Thread.State: RUNNABLE
       at org.apache.hadoop.hdfs.DFSOutputStream.writeChunk(DFSOutputStream.java:425)
       - locked <0x000000020276ba38> (a org.apache.hadoop.hdfs.DFSOutputStream)
       at org.apache.hadoop.fs.FSOutputSummer.writeChecksumChunks(FSOutputSummer.java:214)
       at org.apache.hadoop.fs.FSOutputSummer.flushBuffer(FSOutputSummer.java:165)
       - locked <0x000000020276ba38> (a org.apache.hadoop.hdfs.DFSOutputStream)
       at org.apache.hadoop.fs.FSOutputSummer.flushBuffer(FSOutputSummer.java:146)
       - eliminated <0x000000020276ba38> (a org.apache.hadoop.hdfs.DFSOutputStream)
       at org.apache.hadoop.fs.FSOutputSummer.write1(FSOutputSummer.java:137)
       at org.apache.hadoop.fs.FSOutputSummer.write(FSOutputSummer.java:112)
       - locked <0x000000020276ba38> (a org.apache.hadoop.hdfs.DFSOutputStream)
       at org.apache.hadoop.fs.FSDataOutputStream$PositionCache.write(FSDataOutputStream.java:58)
       at java.io.DataOutputStream.write(DataOutputStream.java:107)
       - locked <0x00000004de9535c8> (a org.apache.hadoop.hdfs.client.HdfsDataOutputStream)
       at java.io.FilterOutputStream.write(FilterOutputStream.java:97)
   ```

Solution

This is a normal phenomenon.

The threads that consume a large number of CPU resources are compaction threads. Some threads invoke the Snappy compression algorithm, and some threads invoke HDFS data writing and reading. Each region has massive sets of data and numerous data files and uses the Snappy compression algorithm. For this reason, the compaction operations consume a large number of CPU resources.

Fault Locating Methods

Step 1 Run the **top** command to check the process with high CPU usage.

Step 2 Check the threads with high CPU usage in the process.

Run the **top -H -p** *<PID>* command to print CPU usage of threads under the process.

Obtain the thread with the highest CPU usage from the query result. You can also obtain the thread by running the following command:

Or run the **ps -mp** *<PID>* **-o THREAD,tid,time | sort -rn** command.

View the command output to obtain the ID of the thread with the highest CPU usage.

Step 3 Obtain the stack of the faulty thread.

The jstack tool is the most effective and reliable tool for locating Java problems.

You can obtain the jstack tool from the **java/bin** directory.

jstack <PID> > allstack.txt

Obtain the process stack and output it to a local file.

Step 4 Convert the thread ID into the hexadecimal format:

printf "%x\n" <PID>

The process ID in the command output is the TID.

Step 5 Run the following command to obtain the TID and output it to a local file:

jstack <PID> | grep <TID> > Onestack.txt

If you want to view the TID in the CLI only, run the following command:

jstack <PID> | grep <TID> -A 30

-A 30 indicates that 30 lines are displayed.

----End

16.8.11 HBase Failed to Started with "FileNotFoundException" in RegionServer Logs

Symptom

HBase fails to start, and the RegionServer stays in the **Restoring** state.

Cause Analysis

1. Check the RegionServer log (**/var/log/Bigdata/hbase/rs/hbase-omm-*XXX*.out**). It is found that the following information is printed:

```
| ERROR | RS_OPEN_REGION-ab-dn01:21302-2 | ABORTING region server ab-dn01,21302,1487663269375: The coprocessor org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService threw java.io.FileNotFoundException: File does not exist: hdfs://hacluster/kylin/kylin_metadata/coprocessor/kylin-coprocessor-1.6.0-SNAPSHOT-0.jar |
org.apache.hadoop.hbase.regionserver.HRegionServer.abort(HRegionServer.java:2123)
java.io.FileNotFoundException: File does not exist: hdfs://hacluster/kylin/kylin_metadata/coprocessor/kylin-coprocessor-1.6.0-SNAPSHOT-0.jar
at org.apache.hadoop.hdfs.DistributedFileSystem$25.doCall(DistributedFileSystem.java:1399)
at org.apache.hadoop.hdfs.DistributedFileSystem$25.doCall(DistributedFileSystem.java:1391)
at org.apache.hadoop.fs.FileSystemLinkResolver.resolve(FileSystemLinkResolver.java:81)
at org.apache.hadoop.hdfs.DistributedFileSystem.getFileStatus(DistributedFileSystem.java:1391)
at org.apache.hadoop.fs.FileUtil.copy(FileUtil.java:340)
at org.apache.hadoop.fs.FileUtil.copy(FileUtil.java:292)
at org.apache.hadoop.fs.FileSystem.copyToLocalFile(FileSystem.java:2038)
at org.apache.hadoop.fs.FileSystem.copyToLocalFile(FileSystem.java:2007)
at org.apache.hadoop.fs.FileSystem.copyToLocalFile(FileSystem.java:1983)
at org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:168)
at org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250)
at org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:224)
at org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:365)
at org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:227)
at org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:783)
at org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:689)
at sun.reflect.GeneratedConstructorAccessor22.newInstance(Unknown Source)
at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
at org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6312)
at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6622)
at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6594)
at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6550)
at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6501)
at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:363)
at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129)
at org.apache.hadoop.hbase.executor.EventHandler.run(EventHandler.java:129)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
at java.lang.Thread.run(Thread.java:745)
```

2. Run the **hdfs** command on the client. It is found that the following file does not exist:

   hdfs://hacluster/kylin/kylin_metadata/coprocessor/kylin-coprocessor-1.6.0-SNAPSHOT-0.jar

3. When configuring the coprocessor for HBase, make sure that the path of the corresponding JAR package is correct. Otherwise, HBase cannot be started.

Solution

Use the Apache Kylin engine to interconnect with MRS and make sure that the JAR file of the Kylin engine exists.

16.8.12 The Number of RegionServers Displayed on the Native Page Is Greater Than the Actual Number After HBase Is Started

Symptom

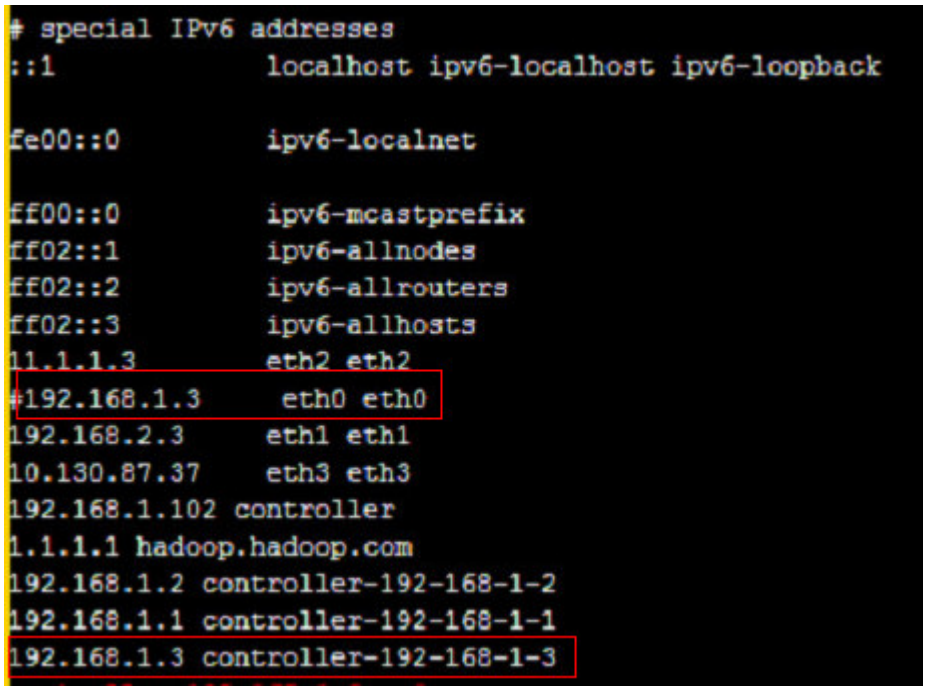
After HBase is started, the number of RegionServers displayed on the HMaster native page is greater than the actual number.

The HMaster native page shows that four RegionServers are online, as shown in the following figure.

ServerName	Start time	Requests Per Second	Num. Regions
controller-192-168-1-1,21302,1494933959261	Tue May 16 19:25:59 CST 2017	0	19
controller-192-168-1-2,21302,1494933957536	Tue May 16 19:25:57 CST 2017	0	24
controller-192-168-1-3,21302,1494933958592	Tue May 16 19:25:58 CST 2017	0	16
eth0,21302,1494933958592	Tue May 16 19:25:58 CST 2017	0	0
Total:4		0	59

Cause Analysis

As shown in the following figure, the hostname of the node in the third row is **controller-192-168-1-3** and that of the fourth row is **eth0**. The two carry the same information reported by RegionServer. Then, log in to the corresponding nodes to check the **/etc/hosts** file. It is found that the same IP address is configured for the two hostnames. For details, see the following figure:



Solution

Log in to the node where RegionServer resides, and modify the **/etc/hosts** file. Make sure that the same IP address can correspond to only one hostname.

16.8.13 RegionServer Instance Is in the Restoring State

Symptom

HBase fails to start, and the RegionServer stays in the **Restoring** state.

Cause Analysis

Check the running log (**/var/log/Bigdata/hbase/rs/hbase-omm-*XXX*.log**) of the abnormal RegionServer instance. It is found that the following information is displayed: **ClockOutOfSyncException..., Reported time is too far out of sync with master**.

```
2017-09-18 11:16:23,636 | FATAL | regionserver21302 | Master rejected startup because clock is out of sync |
org.apache.hadoop.hbase.regionserver.HRegionServer.reportForDuty(HRegionServer.java:2059)
org.apache.hadoop.hbase.ClockOutOfSyncException: org.apache.hadoop.hbase.ClockOutOfSyncException:
Server nl-bi-fi-datanode-24-65,21302,1505726180086 has been rejected; Reported time is too far out of
sync with master.  Time difference of 152109ms > max allowed of 30000ms
at org.apache.hadoop.hbase.master.ServerManager.checkClockSkew(ServerManager.java:354)
...

...
2017-09-18 11:16:23,858 | ERROR | main | Region server exiting |
org.apache.hadoop.hbase.regionserver.HRegionServerCommandLine.start(HRegionServerCommandLine.java:
70)
java.lang.RuntimeException: HRegionServer Aborted
```

This log indicates that the time difference between the abnormal RegionServer instance and the HMaster instance is greater than the allowed time difference 30s (specified by the **hbase.regionserver.maxclockskew** parameter and the default value is **30000 ms**). As a result, the RegionServer instance is abnormal.

Solution

Adjust the node time to ensure that the time difference between nodes is less than 30s.

16.8.14 HBase Failed to Start in a Newly Installed Cluster

Symptom

HBase of a newly installed cluster fails to start. The RegionServer log contains the following error information:

```
2018-02-24 16:53:03,863 | ERROR | regionserver/host3/187.6.71.69:21302 | Master passed us a different hostname to use; was=host3, but now=187-6-71-69 | org.apache.hadoop.hbase.regionserver.HRegionServer.handleReportForDutyResponse(HRegionServer.java:1386)
```

Cause Analysis

In the **/etc/hosts** file, an IP address maps multiple hostnames.

Solution

Step 1 Modify the mapping between the IP address and hostnames in the **/etc/host** file.

Step 2 Restart HBase.

----End

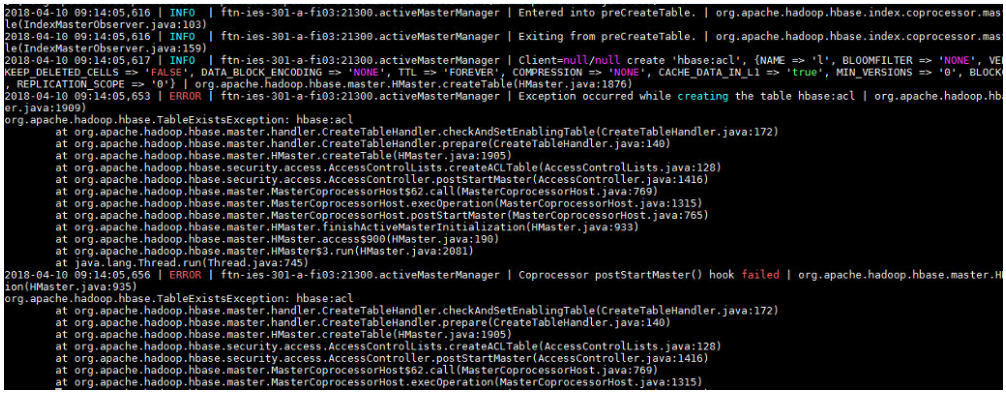
16.8.15 HBase Failed to Start Due to the Loss of the ACL Table Directory

Symptom

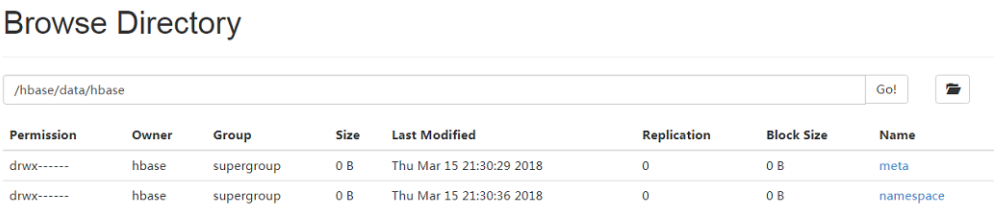
The HBase cluster fails to start.

Cause Analysis

1. Check the HMaster log of HBase. The following error information is displayed:



2. The HBase directory in HDFS is checked, which shows that the ACL table directory is lost.



Solution

Step 1 Stop HBase.

Step 2 Log in to the HBase client as the **hbase** user and run the following command.

Example:

```
hadoop03:~ # source /opt/client/bigdata_env
hadoop03:~ # kinit hbase
Password for hbase@HADOOP.COM:
hadoop03:~ # hbase zkcli
```

Step 3 Delete the ACL table information from the ZooKeeper.

Example:

```
[zk: hadoop01:24002,hadoop02:24002,hadoop03:24002(CONNECTED) 0] deleteall /hbase/table/hbase:acl
[zk: hadoop01:24002,hadoop02:24002,hadoop03:24002(CONNECTED) 0] deleteall /hbase/table-lock/
hbase:acl
```

Step 4 Start HBase.

----End

16.8.16 HBase Failed to Start After the Cluster Is Powered Off and On

Symptom

After the ECS in the cluster is stopped and restarted, HBase fails to start.

Cause Analysis

Check the HMaster run logs. A large number of errors are reported, as shown below:

```
2018-03-26 11:10:54,185 | INFO  | hadoopc1h3,21300,1522031630949_splitLogManager__ChoreService_1 |
total tasks = 1 unassigned = 0 tasks={/hbase/splitWAL/WALs%2Fhadoopc1h1%2C213
02%2C1520214023667-splitting
%2Fhadoopc1h1%252C21302%252C1520214023667.default.1520584926990=last_update =
1522033841041 last_version = 34255 cur_worker_name = hadoopc1h3,21302,
1520943011826 status = in_progress incarnation = 3 resubmits = 3 batch = installed = 1 done = 0 error = 0}
| org.apache.hadoop.hbase.master.SplitLogManager$TimeoutMonitor.chore
(SplitLogManager.java:745)
2018-03-26 11:11:00,185 | INFO  | hadoopc1h3,21300,1522031630949_splitLogManager__ChoreService_1 |
total tasks = 1 unassigned = 0 tasks={/hbase/splitWAL/WALs%2Fhadoopc1h1%2C213
02%2C1520214023667-splitting
%2Fhadoopc1h1%252C21302%252C1520214023667.default.1520584926990=last_update =
1522033841041 last_version = 34255 cur_worker_name = hadoopc1h3,21302,
1520943011826 status = in_progress incarnation = 3 resubmits = 3 batch = installed = 1 done = 0 error = 0}
| org.apache.hadoop.hbase.master.SplitLogManager$TimeoutMonitor.chore
(SplitLogManager.java:745)
```

```
2018-03-26 11:11:06,185 | INFO  | hadoopc1h3,21300,1522031630949_splitLogManager__ChoreService_1 |
total tasks = 1 unassigned = 0 tasks={/hbase/splitWAL/WALs%2Fhadoopc1h1%2C213
02%2C1520214023667-splitting
%2Fhadoopc1h1%252C21302%252C1520214023667.default.1520584926990=last_update =
1522033841041 last_version = 34255 cur_worker_name = hadoopc1h3,21302,
1520943011826 status = in_progress incarnation = 3 resubmits = 3 batch = installed = 1 done = 0 error = 0}
| org.apache.hadoop.hbase.master.SplitLogManager$TimeoutMonitor.chore
(SplitLogManager.java:745)
2018-03-26 11:11:10,787 | INFO  | RpcServer.reader=9,bindAddress=hadoopc1h3,port=21300 | Kerberos
principal name is hbase/hadoop.hadoop.com@HADOOP.COM | org.apache.hadoop.hbase
.ipc.RpcServer$Connection.readPreamble(RpcServer.java:1532)
2018-03-26 11:11:12,185 | INFO  | hadoopc1h3,21300,1522031630949_splitLogManager__ChoreService_1 |
total tasks = 1 unassigned = 0 tasks={/hbase/splitWAL/WALs%2Fhadoopc1h1%2C213
02%2C1520214023667-splitting
%2Fhadoopc1h1%252C21302%252C1520214023667.default.1520584926990=last_update =
1522033841041 last_version = 34255 cur_worker_name = hadoopc1h3,21302,
1520943011826 status = in_progress incarnation = 3 resubmits = 3 batch = installed = 1 done = 0 error = 0}
| org.apache.hadoop.hbase.master.SplitLogManager$TimeoutMonitor.chore
(SplitLogManager.java:745)
2018-03-26 11:11:18,185 | INFO  | hadoopc1h3,21300,1522031630949_splitLogManager__ChoreService_1 |
total tasks = 1 unassigned = 0 tasks={/hbase/splitWAL/WALs%2Fhadoopc1h1%2C213
02%2C1520214023667-splitting
%2Fhadoopc1h1%252C21302%252C1520214023667.default.1520584926990=last_update =
1522033841041 last_version = 34255 cur_worker_name = hadoopc1h3,21302,
1520943011826 status = in_progress incarnation = 3 resubmits = 3 batch = installed = 1 done = 0 error = 0}
| org.apache.hadoop.hbase.master.SplitLogManager$TimeoutMonitor.chore
(SplitLogManager.java:745)
```

The WAL splitting of RegionServer fails when the node is powered on and off.

Solution

Step 1 Stop HBase.

Step 2 Run the **hdfs fsck** command to check the health status of the **/hbase/WALs** file.

hdfs fsck /hbase/WALs

If the following command output is displayed, all files are normal. If any file is abnormal, rectify the fault, and then perform the subsequent operations.

```
The filesystem under path '/hbase/WALs' is HEALTHY
```

Step 3 Back up the **/hbase/WALs** file.

hdfs dfs -mv /hbase/WALs /hbase/WALs_old

Step 4 Run the following command to create the **/hbase/WALs** directory.

hdfs dfs -mkdir /hbase/WALs

Make sure that the permission on the directory is **hbase:hadoop**.

Step 5 Start HBase.

----End

16.8.17 Failed to Import HBase Data Due to Oversized File Blocks

Symptom

Error Message "NotServingRegionException" is displayed when data is imported to HBase.

Cause Analysis

When a block is greater than 2 GB, a read exception occurs during the seek operation of the HDFS. A full GC occurs when data is frequently written to the RegionServer. As a result, the heartbeat between the HMaster and RegionServer becomes abnormal, and the HMaster marks the RegionServer as dead, and the RegionServer is forcibly restarted. After the restart, the servercrash mechanism is triggered to roll back WALs. Currently, the **splitwal** file has reached 2.1 GB and has only one block. As a result, the HDFS seek operation becomes abnormal and the WAL file splitting fails. However, the RegionServer detects that the WAL needs to be split and triggers the splitwal mechanism, causing a loop between WAL splitting and the splitting failure. In this case, the regions on the RegionServer node cannot be brought online, and an exception is thrown indicating that the region is not online when a region on the RegionServer is queried.

Procedure

Step 1 On the right of **HMaster Web UI**, click **HMaster (Active)** to go to the HBase Web UI page.

Step 2 On the **Procedures** page, view the node where the problem occurs.

Step 3 Log in to the faulty node as user **root** and run the **hdfs dfs -ls** command to view all block information.

Step 4 Run the **hdfs dfs -mkdir** command to create a directory for storing faulty blocks.

Step 5 Run the **hdfs dfs -mv** command to move the faulty block to the new directory.

----End

Summary and Suggestions

The following is provided for your reference:

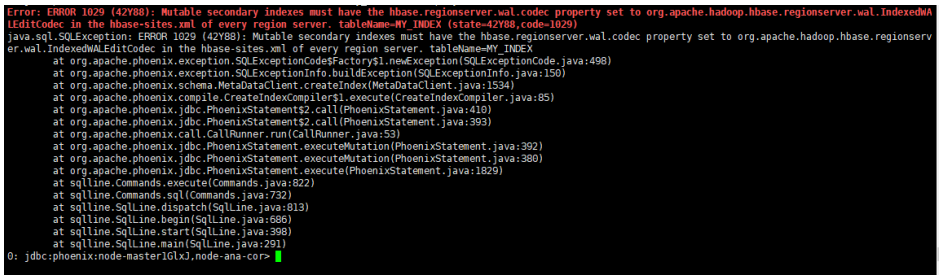
- If data blocks are corrupted, run the **hdfs fsck /tmp -files -blocks -racks** command to check the health information about data blocks.
- If you perform data operations when a region is being split, **NotServingRegionException** is thrown.

16.8.18 Failed to Load Data to the Index Table After an HBase Table Is Created Using Phoenix

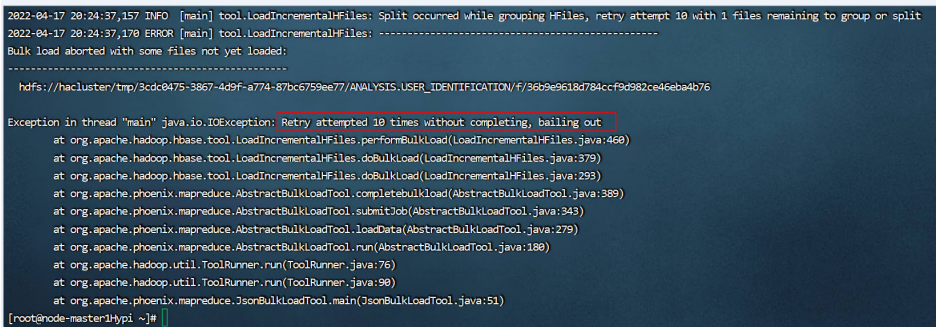
Symptom

A user fails to run commands to load data to the index table after creating an HBase table using Phoenix. The following error information is displayed:

- MRS 2.x or earlier: Mutable secondary indexes must have the hbase.regionserver.wal.codec property set to org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec in the hbase-sites.xml of every region server. tableName=MY_INDEX (state=42Y88,code=1029)



- MRS 3. x or later: Exception in thread "main" java.io.IOException: Retry attempted 10 times without completing, bailing out



Procedure

Step 1 For MRS 2.x or earlier, perform the following operations:

1. Log in to MRS Manager as user **admin**, choose **Services**, and click **HBase**. On the **Service Configuration** tab, select **All** from the **Type** drop-down list, choose **HMaster** > **Customization**, and add a configuration item for parameter **hbase.hmaster.config.expandor** with name **hbase.regionserver.wal.codec** and value **org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec**.
2. Choose **RegionServer** > **Customization**, add a configuration item for parameter **hbase.regionserver.config.expandor** with name **hbase.regionserver.wal.codec** and value **org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec**, and click **Save Configuration**. Then enter the password of the current user and click **OK**.
3. On the **Service Status** page, click **More** and select **Restart Service**. Enter the password of the current user and click **OK** to restart the HBase service.

Step 2 For MRS 3.x or later, perform the following operations:

1. Log in to FusionInsight Manager as user **admin** and choose **Cluster** > **Services** > **HBase**. On the HBase page, choose **Configurations** > **All Configurations** > **RegionServer** > **Customization**. In the right pane, add a configuration item for parameter **hbase.regionserver.config.expandor** with name **hbase.regionserver.wal.codec** and value **org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec**.
2. Choose **HMaster** > **Customization**, and add a configuration item for parameter **hbase.hmaster.config.expandor** with name **hbase.regionserver.wal.codec** and value **org.apache.hadoop.hbase.regionserver.wal.IndexedWALEditCodec**.
3. Click **Save**. In the dialog box that is displayed, click **OK** to save the configuration.
4. On the **Dashboard** page, click **More** and select **Restart Service**. Enter the password of the current user and click **OK** to restart the HBase service.

----End

16.8.19 Failed to Run the hbase shell Command on the MRS Cluster Client

Issue

A user fails to run the **hbase shell** command on the MRS cluster client.

Cause Analysis

- Environment variables have not been configured before the **hbase shell** command is executed.
- The HBase client is not installed in the MRS cluster.

Procedure

Step 1 Log in to the node where the client is installed as user **root**, switch to the client installation directory, and check whether the HBase client is installed.

- If yes, go to **Step 2**.
- If no, download and install the client.

Step 2 Run the following command to set environment variables:

source bigdata_env

Step 3 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the current user. The current user must have the permission to create HBase tables. If Kerberos authentication is disabled for the current cluster, skip this step.

kinit *MRS cluster user*

Step 4 Run the HBase client command.

hbase shell

----End

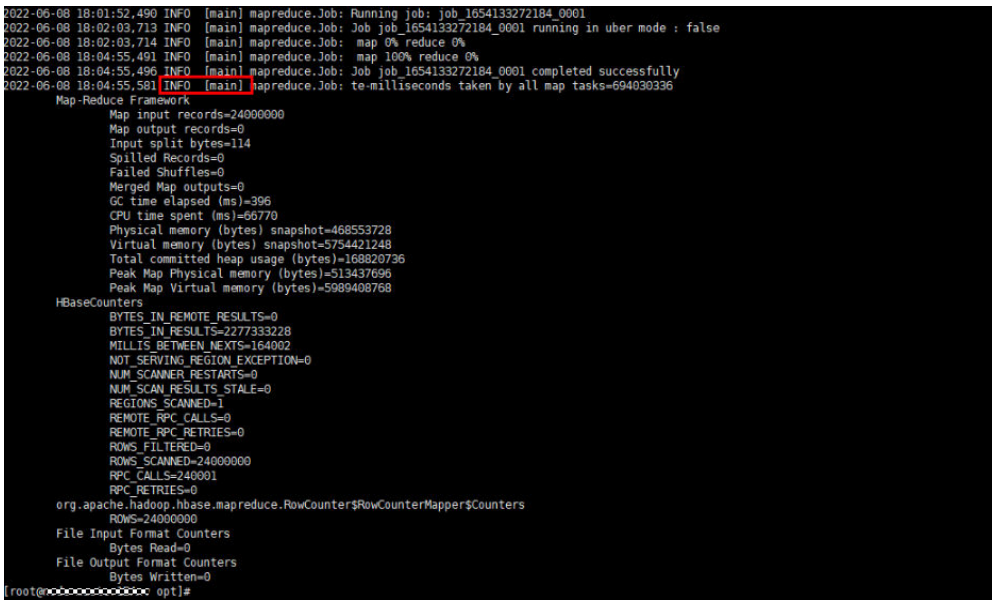
16.8.20 Disordered Information Display on the HBase Shell Client Console Due to Printing of the INFO Information

Issue

When the HBase shell is used, INFO information is printed on the console, causing disordered information display. Before entering a command, the user has to press **Enter** to clean the console. The following figure provides an example.

Cause Analysis

- By default, the log printing function of the HBase client is set to **INFO,console**. Therefore, INFO logs will be printed to the console, affecting the information display on the HBase shell console.
- The HBase client supports many commands, such as **hbase shell**, **hbase hbck**, and **hbase org.apache.hadoop.hbase.mapreduce.RowCounter**, and will support more in the future. The output type of some commands is **INFO**. If **INFO** is disabled, the output of these commands will be lost. For example, the output of **RowCounter** is of the **INFO** type.



Procedure

Step 1 Log in to the node where the HBase client is installed as user **root**.

Step 2 Add the following information to the *HBase client installation directory*/**HBase/component_env** file:

export HBASE_ROOT_LOGGER=INFO,RFA

Logs are exported to log files. If you run the **hbase org.apache.hadoop.hbase.mapreduce.RowCounter** command, you can view the execution result in the *HBase client installation directory*/**HBase/hbase/logs/hbase.log** file.

Step 3 Switch to the HBase client installation directory and run the following commands for the configuration to take effect:

cd *HBase client installation directory*

source HBase/component_env

----End

16.8.21 HBase Failed to Start Due to Insufficient RegionServer Memory

Issue

The HBase service fails to start because the remaining RegionServer memory is insufficient.

Cause Analysis

The troubleshooting process is as follows:

1. Log in to the master node, go to the **/var/log/Bigdata** directory, and search for the HBase log. The log contains error message "connect regionserver timeout".
2. Log in to the RegionServer node in **1** that cannot be connected to HMaster and go to the **/var/log/Bigdata** directory to search for the HBase log. The RegionServer reports error message "error='Cannot allocate memory'(errno=12)".
3. According to the error message in **2**, the startup failure is caused by insufficient RegionServer memory.

Procedure

Step 1 Log in to the RegionServer node where the error is reported and run the following command to check the remaining memory of the node:

free -g

Step 2 Run the **top** command to check the memory usage of the node.

Step 3 Stop the memory-consuming processes (not the processes of the MRS components) as prompted and restart the HBase service.

NOTE

Besides MRS components, jobs on Yarn are allocated to core nodes in the cluster, thereby occupying node memory. If the startup failure is caused by memory-consuming Yarn jobs, you are advised to expand the capacity of core nodes.

----End

16.9 Using HDFS

16.9.1 All NameNodes Become the Standby State After the NameNode RPC Port of HDFS Is Changed

Issue

After the NameNode RPC port is changed on the page and HDFS is restarted, all NameNodes are in the standby state, causing a cluster exception.

Symptom

All NameNodes are in the standby state, causing a cluster exception.

Cause Analysis

After the cluster is installed and started, if the NameNode RPC port is changed, the Zkfc service must be formatted to update node information on ZooKeeper.

Procedure

Step 1 Log in to Manager and stop the HDFS service.

NOTE

Do not stop related services when stopping HDFS.

Step 2 After the services are stopped, log in to the Master node whose RPC port is changed.

NOTE

If the RPC port is changed on both Master nodes, you can log in to either of the Master nodes.

Step 3 Run the **su - omm** command to switch to user **omm**.

NOTE

For a security cluster, run the **kinit hdfs** command for authentication.

Step 4 Run the following command to load the environment variable script to the environment:

cd ${BIGDATA_HOME}/*MRS_X.X.X*_*1_8*_**Zkfc/etc**

source ${BIGDATA_HOME}/*MRS_X.X.X***/install/FusionInsight-Hadoop-3.1.1/hadoop/sbin/exportENV_VARS.sh**

NOTE

In the preceding command, *MRS_X.X.X* and *1_8* vary depending on the actual version.

Step 5 After the loading is complete, run the following command to format the Zkfc:

cd ${HADOOP_HOME}/bin

./hdfs zkfc -formatZK

Step 6 After the formatting is successful, restart HDFS on Manager.

NOTE

If the RPC port of the NameNode is changed, the configuration file must be updated for all clients that have been installed.

----End

16.9.2 An Error Is Reported When the HDFS Client Is Used After the Host Is Connected Using a Public Network IP Address

Issue

When the host is connected using a public network IP address, the HDFS client cannot be used and the message "**-bash: hdfs: command not found**" is displayed when the HDFS is running.

Symptom

When the host is connected using a public network IP address, the HDFS client cannot be used and the message "**-bash: hdfs: command not found**" is displayed when the HDFS is running.

Possible Causes

The environment variables are not set before the user logs in to the Master node and runs the command.

Procedure

Step 1 Log in to any Master node as user **root**.

Step 2 Run the **source /opt/client/bigdata_env** command to configure environment variables.

Step 3 Run the **hdfs** command to use the HDFS client.

----End

16.9.3 Failed to Use Python to Remotely Connect to the Port of HDFS

Issue

Failed to use Python to remotely connect to the port of HDFS.

Symptom

Failed to use Python to remotely connect to port 50070 of HDFS.

Cause Analysis

The default port of open source HDFS is 50070 for versions earlier than 3.0.0 and is 9870 for version 3.0.0 or later. The port used by the user does not match the HDFS version.

Step 1 Log in to the active Master node in the cluster.

Step 2 Run the **su - omm** command to switch to user **omm**.

Step 3 Run the **/opt/Bigdata/om-0.0.1/sbin/queryVersion.sh** command to check the HDFS version in the cluster.

Determine the port number of the open-source component based on the version number.

Step 4 Run the **netstat -anp|grep ${port}** command to check whether the default port number of the component exists.

If it does not exist, the default port number is changed. Change the port to the default port and reconnect to HDFS.

If it exists, contact technical support.

NOTE

- **${port}**: indicates the default port number corresponding to the component version.
- If you have changed the default port number, use the new port number to connect to HDFS. You are advised not to change the default port number.

----End

16.9.4 HDFS Capacity Usage Reaches 100%, Causing Unavailable Upper-layer Services Such as HBase and Spark

Issue

The HDFS capacity usage of the cluster reaches 100%, and the HDFS service status is read-only. As a result, upper-layer services such as HBase and Spark are unavailable.

Symptom

The HDFS capacity usage is 100%, the disk capacity usage is only about 85%, and the HDFS service status is read-only. As a result, upper-layer services such as HBase and Spark are unavailable.

Cause Analysis

Currently, NodeManager and DataNode share data disks. By default, MRS reserves 15% of data disk space for non-HDFS. You can change the percentage of data disk space by setting the HDFS parameter **dfs.datanode.du.reserved.percentage**.

If the HDFS disk usage is 100%, you can set **dfs.datanode.du.reserved.percentage** to a smaller value to restore services and then expand disk capacity.

Procedure

Step 1 Log in to any Master node in the cluster.

Step 2 Run the **source /opt/client/bigdata_env** command to initialize environment variables.

NOTE

If it is a security cluster, run the **kinit -kt <keytab file> <pricipal name>** command for authentication.

Step 3 Run the **hdfs dfs –put ./startDetail.log /tmp** command to check whether HDFS fails to write files.

```
19/05/12 10:07:32 WARN hdfs.DataStreamer: DataStreamer Exception
org.apache.hadoop.ipc.RemoteException(java.io.IOException): File /tmp/startDetail.log._COPYING_ could only be replicated to 0 nodes instead of minReplication (=1).  There are 3 datanode(s) running and no node(s) are excluded in this operation.
```

Step 4 Run the **hdfs dfsadmin -report** command to check the used HDFS capacity. The command output shows that the HDFS capacity usage has reached 100%.

```
Configured Capacity: 5389790579100 (4.90 TB)
Present Capacity: 5067618628404 (4.61 TB)
DFS Remaining: 133350196 (127.17 MB)
DFS Used: 5067485278208 (4.61 TB)
DFS Used%: 100.00%
Under replicated blocks: 10
Blocks with corrupt replicas: 0
Missing blocks: 0
Missing blocks (with replication factor 1): 0
Pending deletion blocks: 0
```

Step 5 When the HDFS capacity usage reaches 100%, change the percentage of data disk space by setting the HDFS parameter **dfs.datanode.du.reserved.percentage**.

1. Go to the service configuration page.
   - MRS Manager: Log in to MRS Manager and choose **Services** > **HDFS** > **Configuration**.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **HDFS** > **Configurations**.
2. Select **All Configurations** and search for **dfs.datanode.du.reserved.percentage** in the search box.
3. Change the value of this parameter to **10**.

Step 6 After the modification, increase the number of disks of the Core node.

----End

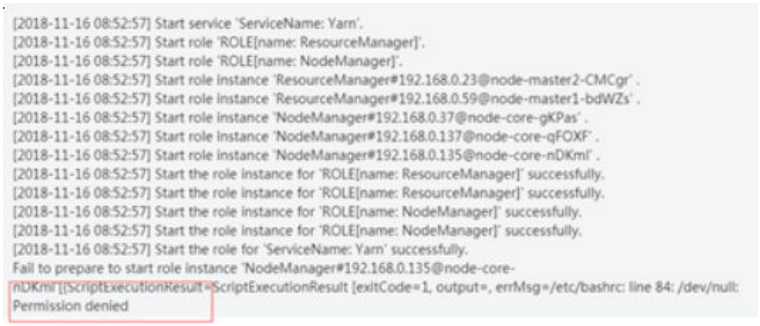
16.9.5 An Error Is Reported During HDFS and Yarn Startup

Issue

An error is reported during HDFS and Yarn startup.

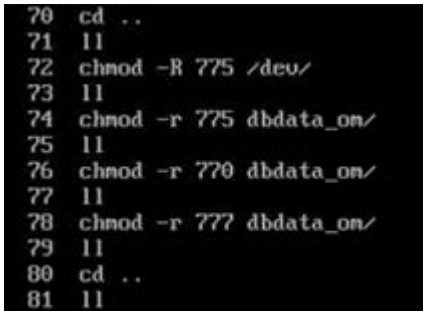
Symptom

HDFS and Yarn fail to be started. The following error information is displayed: **/dev/null Permission denied**



Cause Analysis

The customer changed the permission value of **/dev/null** of the VM system to **775**.



Procedure

Step 1 Log in to any Master node in the cluster as user **root**.

Step 2 After successful login, run the **chmod 666 /dev/null** command to modify the permission value of **/dev/null** to **666**.

Step 3 Run the **ls -al /dev/null** command to check whether the new permission value of **/dev/null** is **666**. If it is not, change the value to **666**.

Step 4 After the modification is successful, restart HDFS and Yarn.

----End

16.9.6 HDFS Permission Setting Error

Issue

When using MRS, a user has the permission to delete or create files in another user's HDFS directory.

Symptom

When using MRS, a user has the permission to delete or create files in another user's HDFS directory.

Cause Analysis

The user has the permission for the **ficommon** group and therefore can perform any operations on the HDFS. You need to remove the user's **ficommon** group permission.

Procedure

Step 1 Log in to the master node in the cluster as user **root**.

Step 2 Run the **id ${Username}** command to check whether the user has the **ficommon** group permission.

If the user has the **ficommon** group permission, go to **Step 3**. If the user does not have the **ficommon** group permission, contact technical support.

NOTE

${Username} indicates the name of the user whose HDFS permission is incorrectly set.

Step 3 Run the **gpasswd -d ${Username} ficommon** command to delete the user's **ficommon** group permission.

NOTE

${Username} indicates the name of the user whose HDFS permission is incorrectly set.

Step 4 Modify parameters on Manager.

MRS Manager (applicable to versions earlier than MRS 3.*x*):

1. Log in to MRS Manager and choose **Services** > **HDFS** > **Service Configuration**.
2. Set **Type** to **All**, enter **dfs.permissions.enabled** in the search box, and change the parameter value to **true**.
3. Click **Save Configuration** and restart the HDFS service.

FusionInsight Manager (applicable to MRS 3.*x* or later):

1. Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **HDFS** > **Configurations** > **All Configurations**.
2. Enter **dfs.permissions.enabled** in the search box and change the value to **true**.
3. After the modification is complete, click **Save** and restart the HDFS service.

MRS console :

1. Log in to the MRS console and choose **Components** > **HDFS** > **Service Configuration**.
2. Set **Type** to **All**, enter **dfs.permissions.enabled** in the search box, and change the parameter value to **true**.

3. Click **Save Configuration** and restart the HDFS service.

----End

16.9.7 A DataNode of HDFS Is Always in the Decommissioning State

Issue

A DataNode of HDFS is in the **Decommissioning** state for a long period of time.

Symptom

A DataNode of HDFS fails to be decommissioned (or the Core node fails to be scaled in), but the DataNode remains in the Decommissioning state.

Cause Analysis

During the decommissioning of a DataNode (or scale-in of the Core node) in HDFS, the decommissioning or scale-in task fails and the blacklist is not cleared because the Master node is restarted or the NodeAgent process exits unexpectedly. In this case, the DataNode remains in the **Decommissioning** state. The blacklist needs to be cleared manually.

Procedure

Step 1 Go to the service instance page.

MRS Manager:

Log in to MRS Manager and choose **Services** > **HDFS** > **Instance**.

FusionInsight Manager:

MRS 3.x or later: Log in to FusionInsight Manager and choose **Cluster** > **Service** > **HDFS** > **Instance**.

Log in to the MRS console and choose **Components** > **HDFS** > **Instances**.

Step 2 Check the HDFS service instance status, locate the DataNode that is in the decommissioning state, and copy the IP address of the DataNode.

Step 3 Log in to the Master1 node and run the **cd ${BIGDATA_HOME}/MRS_*/1_*_NameNode/etc/** command to go to the blacklist directory.

Step 4 Run the **sed -i "/^*IP*$/d" excludeHosts** command to clear the faulty DataNode information from the blacklist. Replace the IP address in the command with the IP address of the faulty DataNode queried in **Step 2**. The IP address cannot contain spaces.

Step 5 If there are two Master nodes, perform **Step 3** and **Step 4** on Master2.

Step 6 Run the following command on the Master1 node to initialize environment variables:

source /opt/client/bigdata_env

Step 7 If Kerberos authentication is enabled for the current cluster, run the following command to authenticate the user. If Kerberos authentication is disabled for the current cluster, skip this step:

kinit *MRS cluster user*

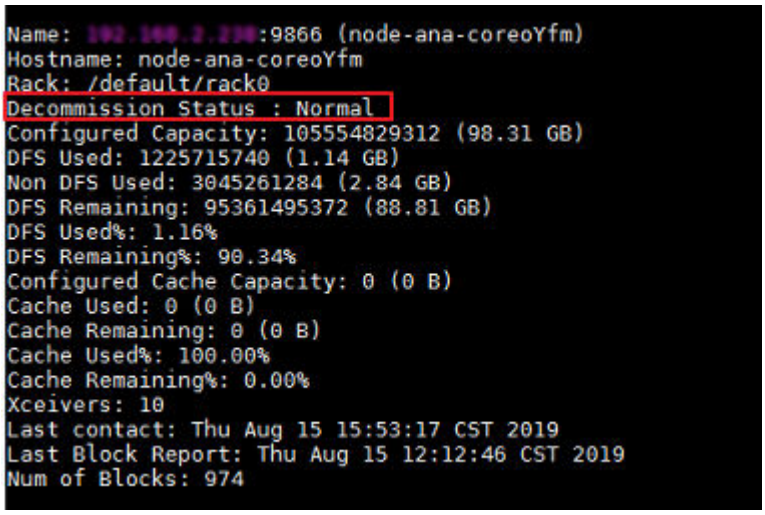
Example: **kinit admin**

Step 8 Run the following command on the Master1 node to update the HDFS blacklist:

hdfs dfsadmin -refreshNodes

Step 9 Run the **hdfs dfsadmin -report** command to check the status of each DataNode. Ensure that the DataNode corresponding to the IP address obtained in has been restored to the **Normal** state.

Figure 16-21 DataNode status



Step 10 Go to the service instance page.

MRS Manager:

Log in to MRS Manager and choose **Services** > **HDFS** > **Instances**.

FusionInsight Manager:

MRS 3.x or later: Log in to FusionInsight Manager and choose **Cluster** > **Service** > **HDFS** > **Instance**.

Log in to the MRS console and choose **Components** > **HDFS** > **Instances**.

Step 11 Select the DataNode instance that is in the decommissioning state and choose **More** > **Restart Instance**.

Step 12 Wait until the restart is complete and check whether the DataNode is restored.

----End

Summary and Suggestions

Do not perform high-risk operations, such as restarting nodes, during decommissioning (or scale-in).

Related Information

None

16.9.8 HDFS Failed to Start Due to Insufficient Memory

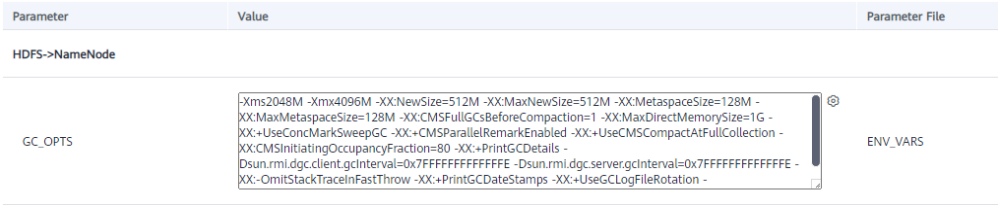
Symptom

After the HDFS service is restarted, HDFS is in the Bad state, the NameNode instance status is abnormal, and the system cannot exit the security mode for a long time.

Cause Analysis

1. In the NameNode run log (**/var/log/Bigdata/hdfs/nn/hadoop-omm-namendoe-XXX.log**), search for **WARN**. It is found that GC takes 63 seconds.
   ```
   2017-01-22 14:52:32,641 | WARN  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@1b39fd82 |
   Detected pause in JVM or host machine (eg GC): pause of approximately 63750ms
   GC pool 'ParNew' had collection(s): count=1 time=0ms
   GC pool 'ConcurrentMarkSweep' had collection(s): count=1 time=63924ms | JvmPauseMonitor.java:189
   ```
2. Analyze the NameNode log **/var/log/Bigdata/hdfs/nn/hadoop-omm-namendoe-XXX.log**. It is found that the NameNode is waiting for block reporting and the total number of blocks is too large. In the following example, the total number of blocks is 36.29 million.
   ```
   2017-01-22 14:52:32,641 | INFO  | IPC Server handler 8 on 25000 | STATE* Safe mode ON.
   The reported blocks 29715437 needs additional 6542184 blocks to reach the threshold 0.9990 of total blocks 36293915.
   ```
3. On Manager, check the **GC_OPTS** parameter of the NameNode:

   Figure 16-22 Checking the GC_OPTS parameter of the NameNode

4. For details about the mapping between the NameNode memory configuration and data volume, see **Table 16-2**.

   Table 16-2 Mapping between NameNode memory configuration and data volume

Number of File Objects	Reference Value
10,000,000	-Xms6G -Xmx6G -XX:NewSize=512M -XX:MaxNewSize=512M
20,000,000	-Xms12G -Xmx12G -XX:NewSize=1G -XX:MaxNewSize=1G
50,000,000	-Xms32G -Xmx32G -XX:NewSize=2G -XX:MaxNewSize=3G
100,000,000	-Xms64G -Xmx64G -XX:NewSize=4G -XX:MaxNewSize=6G

Number of File Objects	Reference Value
200,000,000	-Xms96G -Xmx96G -XX:NewSize=8G -XX:MaxNewSize=9G
300,000,000	-Xms164G -Xmx164G -XX:NewSize=12G -XX:MaxNewSize=12G

Solution

Step 1 Modify the NameNode memory parameter based on the specifications. If the number of blocks is 36 million, change the parameter value to **-Xms32G -Xmx32G -XX:NewSize=2G -XX:MaxNewSize=3G**.

Step 2 Restart a NameNode and check that the NameNode can be started normally.

Step 3 Restart the other NameNode and check that the page status is restored.

----End

16.9.9 A Large Number of Blocks Are Lost in HDFS due to the Time Change Using ntpdate

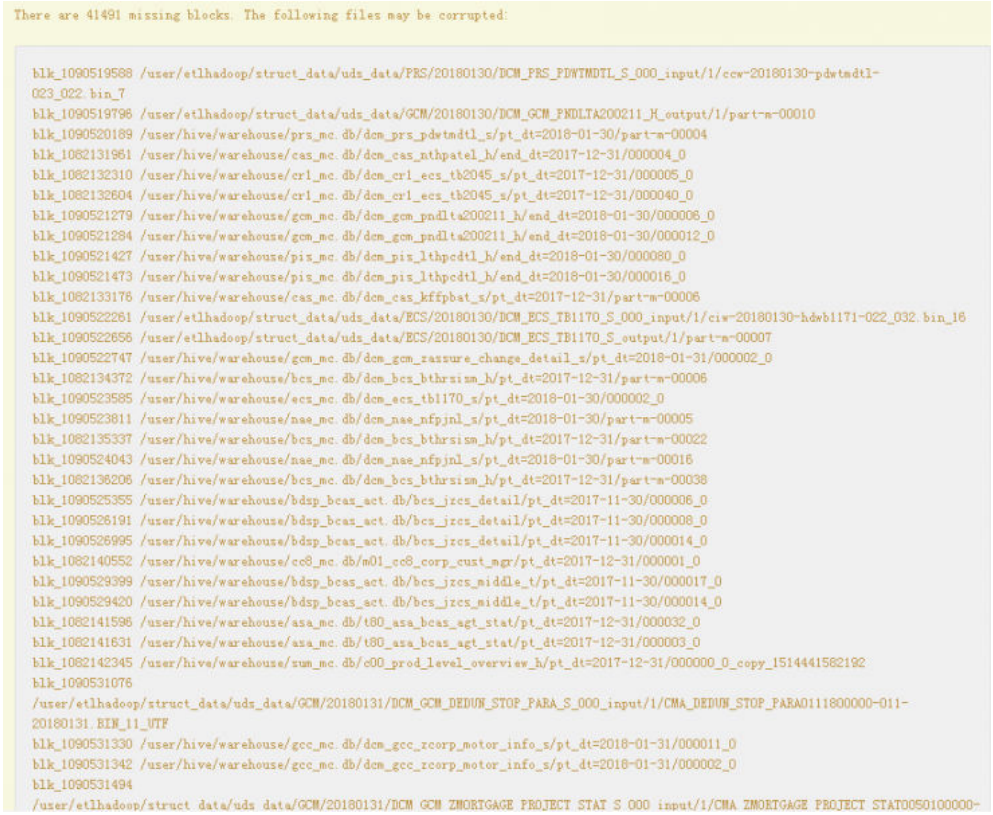
Symptom

1. A user uses **ntpdate** to change the time for a cluster that is not stopped. After the time is changed, HDFS enters the safe mode and cannot be started.
2. After the system exits the safe mode and starts, about 1 TB data is lost during the **hfck** check.

Cause Analysis

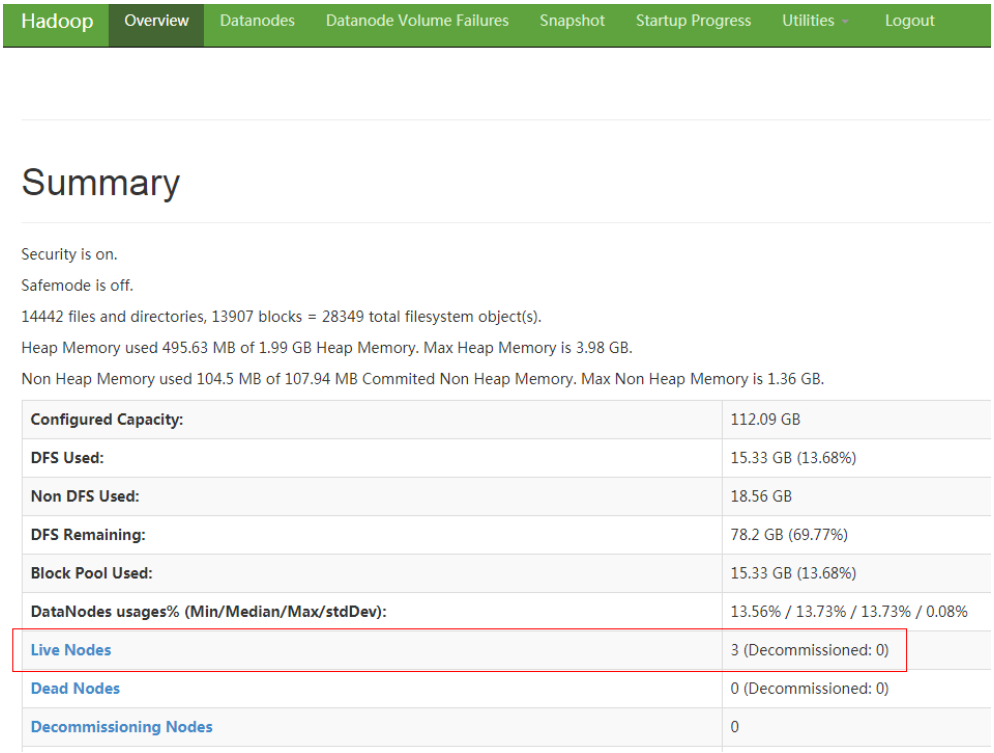
1. A large number of blocks are lost on the native NameNode page.

Figure 16-23 Block loss



2. DataNode information on the native page shows that the number of displayed DataNode nodes is 10 less than that of actual DataNode nodes.

Figure 16-24 Checking the number of DataNodes



3. Check the DataNode run log file **/var/log/Bigdata/hdfs/dn/hadoop-omm-datanode-*hostname*.log**. The following error information is displayed:

   Major error information: Clock skew too great

   Figure 16-25 DateNode run log error

```
    at org.apache.hadoop.ipc.Client.call(Client.java:1486)
    at org.apache.hadoop.ipc.Client.call(Client.java:1447)
    at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:229)
    at com.sun.proxy.$Proxy14.versionRequest(Unknown Source)
    at org.apache.hadoop.hdfs.protocolPB.DatanodeProtocolClientSideTranslatorPB.versionRequest(DatanodeProtocolClientSideTranslatorPB.java:273)
    at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.retrieveNamespaceInfo(BPServiceActor.java:187)
    at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.connectToNNAndHandshake(BPServiceActor.java:237)
    at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.run(BPServiceActor.java:689)
    at java.lang.Thread.run(Thread.java:745)
Caused by: GSSException: No valid credentials provided (Mechanism level: Clock skew too great (37))
    at sun.security.jgss.krb5.Krb5Context.initSecContext(Krb5Context.java:770)
    at sun.security.jgss.GSSContextImpl.initSecContext(GSSContextImpl.java:248)
    at sun.security.jgss.GSSContextImpl.initSecContext(GSSContextImpl.java:179)
    at com.sun.security.sasl.gsskerb.GssKrb5Client.evaluateChallenge(GssKrb5Client.java:192)
    ... 20 more
Caused by: KrbException: Clock skew too great (37)
    at sun.security.krb5.KrbKdcRep.check(KrbKdcRep.java:88)
    at sun.security.krb5.KrbTgsRep.<init>(KrbTgsRep.java:87)
    at sun.security.krb5.KrbTgsReq.getReply(KrbTgsReq.java:259)
    at sun.security.krb5.KrbTgsReq.sendAndGetCreds(KrbTgsReq.java:270)
    at sun.security.krb5.internal.CredentialsUtil.serviceCreds(CredentialsUtil.java:302)
    at sun.security.krb5.internal.CredentialsUtil.acquireServiceCreds(CredentialsUtil.java:120)
    at sun.security.krb5.Credentials.acquireServiceCreds(Credentials.java:458)
    at sun.security.jgss.krb5.Krb5Context.initSecContext(Krb5Context.java:693)
```

Solution

Step 1 Change the time of the 10 DataNodes that cannot be viewed on the native page.

Step 2 On Manager, restart the DataNode instances.

----End

16.9.10 CPU Usage of a DataNode Reaches 100% Occasionally, Causing Node Loss (SSH Connection Is Slow or Fails)

Symptom

The CPU usage of DataNodes is close to 100% occasionally, causing node loss.

Figure 16-26 DataNode CPU usage close to 100%

```
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
60636 omm       20   0 9445m 1.7g  16m S   299  1.3   1952:06 jsvc.exec -Dproc_datanode -outfile /var/log/Bigdata/hdfs/dn/jsvc.out -errfile /var/log/Bigdata/hdfs/dn/jsvc.err -pidfil
02428 ossadm    20   0 18116 3784 1828 R   155  0.0   1:17.63 /opt/tsp/manager/rtsp/python/bin/python /opt/tsp/manager/agent-1.3.10.200/tools/pyscript/sysappctrl.pyc -cmd status -te
02410 ossadm    20   0 55016 8048 2836 R   155  0.0   1:59.80 /opt/tsp/manager/rtsp/python/bin/python /opt/tsp/manager/agent-1.3.10.200/tools/pyscript/watchdog.pyc -cmd status
02412 ossadm    20   0 36752 5912 2340 R   155  0.0   1:50.32 /opt/tsp/manager/rtsp/python/bin/python /opt/tsp/manager/agent-1.3.10.200/tools/pyscript/sysappctrl.pyc -cmd procinfo -
02484 omm       20   0 12800 1476 1124 R   155  0.0   0:10.73 /bin/bash -c /opt/huawei/Bigdata/jdk1.7.0_80//bin/java -server -Xmx1024m -Djava.io.tmpdir=/export/data1/yarn/nm/localdi
02341 ossadm    20   0 57760 8688 3000 R   139  0.0   3:29.41 /opt/tsp/manager/rtsp/python/bin/python /opt/tsp/manager/agent/tools/pyscript/syscollector.pyc sys /opt/tsp/manager/var
02531 omm       20   0 11176  640  468 R   106  0.0   0:04.19 -bash -c echo $OMS_RUN_PATH
02441 root      20   0     0    0    0 S    51  0.0   0:11.87 [kworker/u:0]
```

Cause Analysis

1. A lot of write failure logs exist on DataNodes.

Figure 16-27 DataNode write failure log

```
2015-08-31 11:29:34,184 | ERROR | DataXceiver for client DFSClient_NONMAPREDUCE_1675952887_23 at /192.168.8.40:44514 [Receiving block
BP-125271511-192.168.8.29-1440656260530:blk_1074766997_1034914] | TSP21:25009:DataXceiver error processing WRITE_BLOCK operation  src:
/192.168.8.40:44514 dst: /192.168.8.64:25009 | DataXceiver.java:258
java.io.IOException: Premature EOF from inputStream
        at org.apache.hadoop.io.IOUtils.readFully(IOUtils.java:194)
        at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doReadFully(PacketReceiver.java:213)
        at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doRead(PacketReceiver.java:134)
        at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.receiveNextPacket(PacketReceiver.java:109)
        at org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receivePacket(BlockReceiver.java:446)
        at org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receiveBlock(BlockReceiver.java:707)
        at org.apache.hadoop.hdfs.server.datanode.DataXceiver.writeBlock(DataXceiver.java:748)
        at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.opWriteBlock(Receiver.java:124)
        at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.processOp(Receiver.java:71)
        at org.apache.hadoop.hdfs.server.datanode.DataXceiver.run(DataXceiver.java:240)
        at java.lang.Thread.run(Thread.java:745)
2015-08-31 11:29:35,147 | INFO  | DataXceiver for client DFSClient_NONMAPREDUCE_-402997805_1 at /192.168.8.30:59449 [Sending block BP-
125271511-192.168.8.29-1440656260530:blk_1074181856_446655] | src: /192.168.8.64:25009, dest: /192.168.8.30:59449, bytes: 16826, op:
HDFS_READ, cliID: DFSClient_NONMAPREDUCE_-402997805_1, offset: 0, srvID: 9d1d30a5-046d-438b-83c9-2c6c54c6bd12, blockid: BP-125271511-
192.168.8.29-1440656260530:blk_1074181856_446655, duration: 78832 | BlockSender.java:738
2015-08-31 11:29:35,269 | INFO  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg
GC): pause of approximately 7480ms
No GCs detected | JvmPauseMonitor.java:172
2015-08-31 11:29:36,985 | INFO  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg
GC): pause of approximately 1215ms
No GCs detected | JvmPauseMonitor.java:172
2015-08-31 11:29:43,067 | INFO  | DataXceiver for client DFSClient_NONMAPREDUCE_1675952887_23 at /192.168.8.33:35530 [Receiving block
BP-125271511-192.168.8.29-1440656260530:blk_1074767006_1034923] | Exception for BP-125271511-192.168.8.29-
1440656260530:blk_1074767006_1034923 | BlockReceiver.java:742
java.io.IOException: Premature EOF from inputStream
```

2. A large number of files are written in a short time, causing insufficient DataNode memory.

Figure 16-28 Insufficient DataNode memory

```
Line 153101: 2015-08-31 11:24:29,313 | INFO  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg GC): pause of approximately 1199ms
Line 153132: 2015-08-31 11:24:42,689 | WARN  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg GC): pause of approximately 11273ms
Line 153135: 2015-08-31 11:24:45,810 | INFO  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg GC): pause of approximately 1005ms
Line 153138: 2015-08-31 11:24:49,801 | INFO  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg GC): pause of approximately 1067ms
Line 153155: 2015-08-31 11:25:10,167 | WARN  | org.apache.hadoop.util.JvmPauseMonitor$Monitor@551bd2a0 | Detected pause in JVM or host machine (eg GC): pause of approximately 12323ms
```

Solution

Step 1 Check DataNode memory configuration and whether the remaining server memory is sufficient.

Step 2 Increase DataNode memory and restart the DataNode.

----End

16.9.11 Manually Performing Checkpoints When a NameNode Is Faulty for a Long Time

Symptom

If the standby NameNode is faulty for a long time, a large amount of editlog will be accumulated. In this case, if the HDFS or active NameNode is restarted, the active NameNode reads a large amount of unmerged editlog. As a result, the HDFS or active NameNode takes a long time to restart and even fails to restart.

Cause Analysis

The standby NameNode periodically combines editlog files and generates the fsimage file. This process is called checkpoint. After the fsimage file is generated, the standby NameNode transfers it to the active NameNode.

NOTE

As the standby NameNode periodically combines editlog files, it cannot combine them when it becomes abnormal. As a result, the active NameNode needs to load many editlog files during its next startup, which occupies much memory and takes a long time.

The period of metadata combination is determined by the following parameters. If the NameNode runs for 30 minutes or one million counts of operations are performed on HDFS, the checkpoint is implemented.

- **dfs.namenode.checkpoint.period**: specifies the checkpoint period. The default value is **1800s**.
- dfs.namenode.checkpoint.txns: specifies the times of operations for triggering the checkpoint execution. The default value is **1000000**.

Solution

Before restarting the HDFS or active NameNode, perform checkpoint manually to merge metadata of the active NameNode.

Step 1 Stop workloads.

Step 2 Obtain the hostname of the active NameNode.

Step 3 Run the following commands on the client:

source /opt/client/bigdata_env

kinit *Component user*

Note: Replace **/opt/client** with the actual installation path of the client.

Step 4 Run the following command to enable the safe mode for the active NameNode (replace **linux22** with the hostname of the active NameNode):

hdfs dfsadmin -fs linux22:25000 -safemode enter

```
linux16:/opt/fi_client #  hdfs dfsadmin -fs linux22:25000   -safemode enter
17/04/26 18:38:30 WARN fs.FileSystem: "linux22:25000" is a deprecated filesystem name. Use "hdfs://linux22:25000/" instead.
17/04/26 18:38:32 INFO hdfs.PeerCache: SocketCache disabled.
Safe mode is ON
```

Step 5 Run the following command to merge editlog on the active NameNode:

hdfs dfsadmin -fs linux22:25000 -saveNamespace

```
linux16:/opt/fi_client #  hdfs dfsadmin -fs linux22:25000   -saveNamespace
17/04/26 18:38:54 WARN fs.FileSystem: "linux22:25000" is a deprecated filesystem name. Use "hdfs://linux22:25000/" instead.
17/04/26 18:38:56 INFO hdfs.PeerCache: SocketCache disabled.
Save namespace successful
```

Step 6 Run the following command to make the active NameNode exit the safe mode:

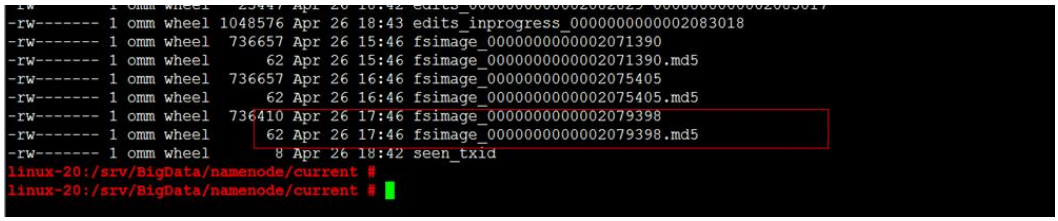
hdfs dfsadmin -fs linux22:25000 -safemode leave

```
linux16:/opt/fi_client #  hdfs dfsadmin -fs linux22:25000  -safemode leave
17/04/26 18:39:07 WARN fs.FileSystem: "linux22:25000" is a deprecated filesystem name. Use "hdfs://linux22:25000/" instead.
17/04/26 18:39:09 INFO hdfs.PeerCache: SocketCache disabled.
Safe mode is OFF
```

Step 7 Check whether the combination is complete.

cd /srv/BigData/namenode/current

Check whether the time of the first generated fsimage is the current time. If yes, the combination is complete.



----End

16.9.12 Common File Read/Write Faults

Symptom

When a user performs a write operation on HDFS, the message "Failed to place enough replicas:expected…" is displayed.

Cause Analysis

- The data receiver of the DataNode is unavailable.

  The DataNode log is as follows:

  ```
  2016-03-17 18:51:44,721 | WARN |
  org.apache.hadoop.hdfs.server.datanode.DataXceiverServer@5386659f |
  hadoopc1h2:25009:DataXceiverServer: | DataXceiverServer.java:158
  java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
  at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:140)
  at java.lang.Thread.run(Thread.java:745)
  ```

- The disk space configured for the DataNode is insufficient.
- DataNode heartbeats are delayed.

Solution

- If the DataNode data receiver is unavailable, add the value of the HDFS parameter **dfs.datanode.max.transfer.threads** on Manager.
- If disk space or CPU resources are insufficient, add DataNodes or ensure that disk space and CPU resources are available.
- If the network is faulty, ensure that the network is available.

16.9.13 Maximum Number of File Handles Is Set to a Too Small Value, Causing File Reading and Writing Exceptions

Symptom

The maximum number of file handles is set to a too small value, causing insufficient file handles. Writing files to HDFS is slow or file writing fails.

Cause Analysis

1. The DataNode log **/var/log/Bigdata/hdfs/dn/hadoop-omm-datanode-XXX.log** contains exception information "java.io.IOException: Too many open files."

   ```
   2016-05-19 17:18:59,126 | WARN |
   org.apache.hadoop.hdfs.server.datanode.DataXceiverServer@142ff9fa |
   YSDN12:25009:DataXceiverServer: |
   ```

```
org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:160)
java.io.IOException: Too many open files
at sun.nio.ch.ServerSocketChannelImpl.accept0(Native Method)
at sun.nio.ch.ServerSocketChannelImpl.accept(ServerSocketChannelImpl.java:241)
at sun.nio.ch.ServerSocketAdaptor.accept(ServerSocketAdaptor.java:100)
at org.apache.hadoop.hdfs.net.TcpPeerServer.accept(TcpPeerServer.java:134)
at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:137)
at java.lang.Thread.run(Thread.java:745)
```

2. The error indicates insufficient file handles. File handles cannot be opened and data is written to other DataNodes. As a result, writing files is slow or fails.

Solution

Step 1 Run the **ulimit -a** command to check the maximum number of file handles set for the involved node. If the value is small, change it to **640000**.

Figure 16-29 Check the number of file handles.

```
[omm@189-39-150-167 ~]$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimited
pending signals                 (-i) 256551
max locked memory       (kbytes, -l) 64
max memory size         (kbytes, -m) unlimited
open files                      (-n) 640000
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 10240
cpu time               (seconds, -t) unlimited
max user processes              (-u) 60000
virtual memory          (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Step 2 Run the **vi /etc/security/limits.d/90-nofile.conf** command to edit this file. Set the number of file handles to **64000**. If the file does not exist, create one and modify the file as follows:

Figure 16-30 Changing the number of file handles

```
*          hard    nofile    640000
*          soft    nofile    640000
~
```

Step 3 Open another terminal. Run the **ulimit -a** command to check whether the modification is successful. If the modification fails, perform the preceding operations again.

Step 4 Restart the DataNode instance on Manager.

----End

16.9.14 A Client File Fails to Be Closed After Data Writing

Symptom

A client file fails to be closed after data is written to the file. A message is displayed indicating that the data block does not have enough replicas.

Client log:

```
2015-05-27 19:00:52.811 [pool-2-thread-3] ERROR: /tsp/nedata/collect/UGW/ugwufdr/
20150527/10/6_20150527105000_20150527105500_SR5S14_1432723806338_128_11.pkg.tmp143272380633
8 close hdfs sequence file fail (SequenceFileInfoChannel.java:444)
java.io.IOException: Unable to close file because the last block does not have enough number of replicas.
at org.apache.hadoop.hdfs.DFSOutputStream.completeFile(DFSOutputStream.java:2160)
at org.apache.hadoop.hdfs.DFSOutputStream.close(DFSOutputStream.java:2128)
at org.apache.hadoop.fs.FSDataOutputStream$PositionCache.close(FSDataOutputStream.java:70)
at org.apache.hadoop.fs.FSDataOutputStream.close(FSDataOutputStream.java:103)
at com.xxx.pai.collect2.stream.SequenceFileInfoChannel.close(SequenceFileInfoChannel.java:433)
at com.xxx.pai.collect2.stream.SequenceFileWriterToolChannel
$FileCloseTask.call(SequenceFileWriterToolChannel.java:804)
at com.xxx.pai.collect2.stream.SequenceFileWriterToolChannel
$FileCloseTask.call(SequenceFileWriterToolChannel.java:792)
at java.util.concurrent.FutureTask.run(FutureTask.java:262)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1145)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:615)
at java.lang.Thread.run(Thread.java:745)
```

Cause Analysis

1. The HDFS client starts to write blocks.

   For example, the HDFS client starts to write **/20150527/10/6_20150527105000_20150527105500_SR5S14_1432723806338_128_11.pkg.tmp1432723806338 at 2015-05-27 18:50:24,232**. The allocated block is **blk_1099105501_25370893**:

   ```
   2015-05-27 18:50:24,232 | INFO  | IPC Server handler 30 on 25000 | BLOCK* allocateBlock: /
   20150527/10/6_20150527105000_20150527105500_SR5S14_1432723806338_128_11.pkg.tmp1432723
   806338. BP-1803470917-192.168.57.33-1428597734132
   blk_1099105501_25370893{blockUCState=UNDER_CONSTRUCTION, primaryNodeIndex=-1,
   replicas=[ReplicaUnderConstruction[[DISK]DS-b2d7b7d0-f410-4958-8eba-6deecbca2f87:NORMAL|
   RBW], ReplicaUnderConstruction[[DISK]DS-76bd80e7-ad58-49c6-bf2c-03f91caf750f:NORMAL|RBW]]}
   |
   org.apache.hadoop.hdfs.server.namenode.FSNamesystem.saveAllocatedBlock(FSNamesystem.java:3166
   )
   ```

2. After the writing is complete, the HDFS client invokes **fsync**:

   ```
   2015-05-27 19:00:22,717 | INFO  | IPC Server handler 22 on 25000 | BLOCK* fsync:
   20150527/10/6_20150527105000_20150527105500_SR5S14_1432723806338_128_11.pkg.tmp1432723
   806338 for DFSClient_NONMAPREDUCE_-120525246_15 |
   org.apache.hadoop.hdfs.server.namenode.FSNamesystem.fsync(FSNamesystem.java:3805)
   ```

3. The HDFS client invokes **close** to close the file. After receiving the close request from the client, the NameNode uses the checkFileProgress function to check the completion status of the last block and closes the file only when enough DataNodes report that the last block is complete:

   ```
   2015-05-27 19:00:27,603 | INFO  | IPC Server handler 44 on 25000 | BLOCK* checkFileProgress:
   blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
   replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
   RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
   has not reached minimal replication 1 |
   org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkFileProgress(FSNamesystem.java:3197)
   2015-05-27 19:00:28,005 | INFO  | IPC Server handler 45 on 25000 | BLOCK* checkFileProgress:
   blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
   replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
   RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
   ```

```
has not reached minimal replication 1 |
org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkFileProgress(FSNamesystem.java:3197)
2015-05-27 19:00:28,806 | INFO  | IPC Server handler 63 on 25000 | BLOCK* checkFileProgress:
blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
has not reached minimal replication 1 |
org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkFileProgress(FSNamesystem.java:3197)
2015-05-27 19:00:30,408 | INFO  | IPC Server handler 43 on 25000 | BLOCK* checkFileProgress:
blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
has not reached minimal replication 1 |
org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkFileProgress(FSNamesystem.java:3197)
2015-05-27 19:00:33,610 | INFO  | IPC Server handler 37 on 25000 | BLOCK* checkFileProgress:
blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
has not reached minimal replication 1 |
org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkFileProgress(FSNamesystem.java:3197)
2015-05-27 19:00:40,011 | INFO  | IPC Server handler 37 on 25000 | BLOCK* checkFileProgress:
blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
has not reached minimal replication 1 |
org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkFileProgress(FSNamesystem.java:3197)
```

4. The NameNode prints **CheckFileProgress** multiple times because the HDFS client retries to close the file for several times. The file closing fails because the block status is not complete. The number of retries is determined by the **dfs.client.block.write.locateFollowingBlock.retries** parameter. The default value is **5**. Therefore, **CheckFileProgress** is printed six times in the NameNode log.
5. After 0.5 seconds, the DataNodes report that the block has been successfully written.

   ```
   2015-05-27 19:00:40,608 | INFO  | IPC Server handler 60 on 25000 | BLOCK* addStoredBlock:
   blockMap updated: 192.168.10.21:25009 is added to
   blk_1099105501_25370893{blockUCState=COMMITTED, primaryNodeIndex=-1,
   replicas=[ReplicaUnderConstruction[[DISK]DS-ef5fd3c9-5088-4813-ae9a-34a0714ec3a3:NORMAL|
   RBW], ReplicaUnderConstruction[[DISK]DS-f863e30f-ce5b-48cc-9cca-72f64c558adc:NORMAL|RBW]]}
   size 11837530 |
   org.apache.hadoop.hdfs.server.blockmanagement.BlockManager.logAddStoredBlock(BlockManager.java
   :2393)
   2015-05-27 19:00:48,297 | INFO  | IPC Server handler 37 on 25000 | BLOCK* addStoredBlock:
   blockMap updated: 192.168.10.10:25009 is added to blk_1099105501_25370893 size 11837530 |
   org.apache.hadoop.hdfs.server.blockmanagement.BlockManager.logAddStoredBlock(BlockManager.java
   :2393)
   ```

6. The block write success notification is delayed because of network bottlenecks or CPU bottlenecks.
7. If close is invoked again or the number of file closing retries increases, a closing success message will be displayed. You are advised to increase the value of **dfs.client.block.write.locateFollowingBlock.retries**. The default parameter value is 5 and retry intervals are 400 ms, 800 ms, 1600 ms, 3200 ms, 6400 ms, and 12800 ms. Therefore, the result of the close function can be returned after a maximum of 25.2 seconds.

Solution

Step 1 Solution:

Set the value of **dfs.client.block.write.locateFollowingBlock.retries** to **6**. The retry intervals are 400 ms, 800 ms, 1600 ms, 3200 ms, 6400 ms, and 12800 ms.

Therefore, the result of the close function can be returned after a maximum of 50.8 seconds.

----End

Remarks

Generally, this fault occurs when the cluster workload is heavy. Adjusting the parameter can only temporarily avoid the fault. You are advised to reduce the cluster workload, for example, do not allocate all CPU resources to MapReduce.

16.9.15 File Fails to Be Uploaded to HDFS Due to File Errors

Symptom

The **hadoop dfs -put** command is used to copy local files to HDFS.

After some files are uploaded, an error occurs. The size of the temporary files no long changes on the native NameNode page.

Cause Analysis

1. Check the NameNode log **/var/log/Bigdata/hdfs/nn/hadoop-omm-namenode-*hostname*.log**. It is found that the file is being written until a failure occurs.

   ```
   2015-07-13 10:05:07,847 | WARN  | org.apache.hadoop.hdfs.server.namenode.LeaseManager
   $Monitor@36fea922 | DIR* NameSystem.internalReleaseLease: Failed to release lease for file /hive/
   order/OS_ORDER._8.txt._COPYING_. Committed blocks are waiting to be minimally replicated. Try
   again later. | FSNamesystem.java:3936
   2015-07-13 10:05:07,847 | ERROR | org.apache.hadoop.hdfs.server.namenode.LeaseManager
   $Monitor@36fea922 | Cannot release the path /hive/order/OS_ORDER._8.txt._COPYING_ in the lease
   [Lease.  Holder: DFSClient_NONMAPREDUCE_-1872896146_1, pendingcreates: 1] |
   LeaseManager.java:459
   org.apache.hadoop.hdfs.protocol.AlreadyBeingCreatedException: DIR*
   NameSystem.internalReleaseLease: Failed to release lease for file /hive/order/
   OS_ORDER._8.txt._COPYING_. Committed blocks are waiting to be minimally replicated. Try again
   later.
   at FSNamesystem.internalReleaseLease(FSNamesystem.java:3937)
   ```

2. Root cause: The uploaded files are damaged.
3. Verification: The cp or scp operation fails to be performed for the copied files. Therefore, the files are damaged.

Solution

Step 1 Upload normal files.

----End

16.9.16 After dfs.blocksize Is Configured and Data Is Put, Block Size Remains Unchanged

Symptom

After **dfs.blocksize** is set to **268435456** on the interface and data is put, the original block size keeps unchanged.

Cause Analysis

The **dfs.blocksize** value in the **hdfs-site.xml** file of the client is not changed, and the value prevails.

Solution

Step 1 Ensure that the **dfs.blocksize** value is a multiple of 512.

Step 2 Download a client or modify the client configuration.

Step 3 **dfs.blocksize** is configured on the client and is subject to the client. Otherwise, the value configured on the server prevails.

----End

16.9.17 Failed to Read Files, and "FileNotFoundException" Is Displayed

Symptom

In MapReduce tasks, all Map tasks are successfully executed, but Reduce tasks fail. The error message "FileNotFoundException...No lease on...File does not exist" is displayed in the logs.

```
Error: org.apache.hadoop.ipc.RemoteException(java.io.FileNotFoundException): No lease on /user/sparkhive/warehouse/daas/dsp/output/_temporary/1/_temporary/attempt_1479799053892_17075_r_000007_0/part-r-00007 (inode 6501287): File does not exist. Holder DFSClient_attempt_1479799053892_17075_r_000007_0_-1463597952_1 does not have any open files.
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkLease(FSNamesystem.java:3350)
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.completeFileInternal(FSNamesystem.java:3442)
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.completeFile(FSNamesystem.java:3409)
at org.apache.hadoop.hdfs.server.namenode.NameNodeRpcServer.complete(NameNodeRpcServer.java:789)
```

Cause Analysis

"FileNotFoundException...No lease on...File does not exist" indicates that the file is deleted during the operation.

1. Search for the file name in the NameNode audit log of HDFS (**/var/log/Bigdata/audit/hdfs/nn/hdfs-audit-namenode.log** of the active NameNode) to confirm the creation time of the file.
2. Search the NameNode audit logs that are generated within the time range from the file creation to the time of exception occurrence and determine whether the file is deleted or moved to another directory.
3. If the file is not deleted or moved, the parent directory of the file may be deleted or moved. You need to search the upper-layer directory. In this example, the parent directory of the file's parent directory is deleted.

   ```
   2017-05-31 02:04:08,286 | INFO  | IPC Server handler 30 on 25000 | allowed=true ugi=appUser@HADOOP.COM (auth:TOKEN) ip=/192.168.1.22 cmd=delete src=/user/sparkhive/warehouse/daas/dsp/output/_temporary dst=null perm=null proto=rpc | FSNamesystem.java:8189
   ```

NOTE

- The preceding log indicates that the **appUser** user of the 192.168.1.22 node deletes **/user/sparkhive/warehouse/daas/dsp/output/_temporary**.
- Run the **zgrep "***file name***" *.zip** command to search for the contents of the .zip package.

Solution

Step 1 Check the service to find out why the file or the parent directory of the file is deleted.

----End

16.9.18 Failed to Write Files to HDFS, and "item limit of / is exceeded" Is Displayed

Symptom

The client or upper-layer component logs indicate that a file fails to be written to a directory on HDFS. The error information is as follows:

The directory item limit of /tmp is exceeded: limit=5 items=5.

Cause Analysis

1. The run log file **/var/log/Bigdata/hdfs/nn/hadoop-omm-namenode-XXX.log** of the client or NameNode contains error information "The directory item limit of /tmp is exceeded:." The error message indicates that the number of files in the **/tmp** directory exceeds 1048576.
   2018-03-14 11:18:21,625 | WARN | IPC Server handler 62 on 25000 | DIR* NameSystem.startFile: /tmp/test.txt The directory item limit of /tmp is exceeded: limit=1048576 items=1048577 | FSNamesystem.java:2334
2. The **dfs.namenode.fs-limits.max-directory-items** parameter specifies the maximum number of directories or files that are not in recursion relationship in a single directory. The default value is **1048576**. The value ranges from 1 to 6400000.

Solution

Step 1 Check whether it is normal that the directory contains more than one million files that are not in recursion relationship. If it is normal, increase the value of the HDFS parameter **dfs.namenode.fs-limits.max-directory-items** and restart the HDFS NameNode for the modification to take effect.

Step 2 If it is abnormal, delete unnecessary files.

----End

16.9.19 Adjusting the Log Level of the Shell Client

- **Temporary adjustment**: After the Shell client window is closed, the log is restored to the default value.

a. Run the **export HADOOP_ROOT_LOGGER** command to adjust the log level of the client.

b. Run the **export HADOOP_ROOT_LOGGER=***log level***,console** command to adjust the log level of the Shell client.

   Run the **export HADOOP_ROOT_LOGGER=DEBUG,console** command to adjust the log level to **Debug**.

   Run the **export HADOOP_ROOT_LOGGER=ERROR,console** command to adjust the log level to **Error**.

- **Permanent adjustment**

  a. Add **export HADOOP_ROOT_LOGGER**=*log level***,console** to the HDFS client's environment variable configuration file **/opt/client/HDFS/component_env** (replace **/opt/client** with the actual client path).

  b. Run the **source /opt/client/bigdata_env** command.

  c. Run the command on the client again.

16.9.20 File Read Fails, and "No common protection layer" Is Displayed

Symptom

HDFS fails to be operated on the Shell client or other clients, and the error message "No common protection layer between client and server" is displayed.

Running any **hadoop** command, such as **hadoop fs -ls /**, on a node outside the cluster fails. The bottom-layer error message is displayed stating "No common protection layer between client and server."

```
2017-05-13 19:14:19,060 | ERROR | [pool-1-thread-1] |  Server startup failure  |
org.apache.sqoop.core.SqoopServer.initializeServer(SqoopServer.java:69)
org.apache.sqoop.common.SqoopException: MAPRED_EXEC_0028:Failed to operate HDFS - Failed to get the
file /user/loader/etl_dirty_data_dir status
        at org.apache.sqoop.job.mr.HDFSClient.fileExist(HDFSClient.java:85)
...
        at java.lang.Thread.run(Thread.java:745)
Caused by: java.io.IOException: Failed on local exception: java.io.IOException: Couldn't setup connection for
loader/hadoop@HADOOP.COM to loader37/10.162.0.37:25000; Host Details : local host is:
"loader37/10.162.0.37"; destination host is: "loader37":25000;
        at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:776)
...
        ... 10 more
Caused by: java.io.IOException: Couldn't setup connection for loader/hadoop@HADOOP.COM to
loader37/10.162.0.37:25000
        at org.apache.hadoop.ipc.Client$Connection$1.run(Client.java:674
        ... 28 more
Caused by: javax.security.sasl.SaslException: No common protection layer between client and server
        at com.sun.security.sasl.gsskerb.GssKrb5Client.doFinalHandshake(GssKrb5Client.java:251)
...
        at org.apache.hadoop.ipc.Client$Connection.setupIOstreams(Client.java:720)
```

Cause Analysis

1. The RPC protocol is used for data transmission between the client and server of HDFS. The protocol has multiple encryption modes and the hadoop.rpc.protection parameter specifies the mode to use.

2. If the value of the **hadoop.rpc.protection** parameter on the client is different from that on the server, the "No common protection layer between client and server" error is reported.

   NOTE

   hadoop.rpc.protection indicates that data can be transmitted between nodes in any of the following modes:

   - **privacy**: Data is transmitted after authentication and encryption. This mode reduces the performance.
   - **authentication**: Data is transmitted after authentication without encryption. This mode ensures performance but has security risks.
   - **integrity**: Data is transmitted without encryption or authentication. To ensure data security, exercise caution when using this mode.

Solution

Step 1 Download the client again. If the client is an application, update the configuration file in the application.

----End

16.9.21 Failed to Write Files Because the HDFS Directory Quota Is Insufficient

Symptom

After quota is set for a directory, writing files to the directory fails. The "The DiskSpace quota of /tmp/tquota2 is exceeded" error message is displayed.

```
[omm@189-39-150-115 client]$ hdfs dfs -put switchuser.py  /tmp/tquota2
put: The DiskSpace quota of /tmp/tquota2 is exceeded: quota = 157286400 B = 150 MB but diskspace consumed = 402653184 B = 384 MB
```

Possible Causes

The remaining space configured for the directory is less than the space required for writing files.

Cause Analysis

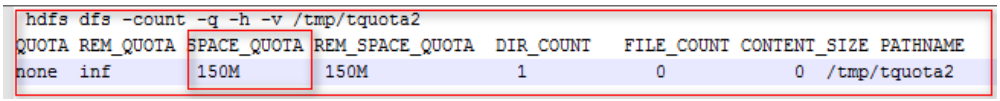
1. The HDFS supports setting the quota for a specific directory, that is, the maximum space occupied by files in a directory can be set. For example, the following command is used to set a maximum of 150 MB files to be written to the **/tmp/tquota** directory. (Space = Block size x Number of copies)

   hadoop dfsadmin -setSpaceQuota 150M /tmp/tquota2

2. Run the following command to check the configured quota for the directory. **SPACE_QUOTA** is the configured space quota, and **REM_SPACE_QUOTA** is the remaining space.

   hdfs dfs -count -q -h -v /tmp/tquota2

   Figure 16-31 Viewing the quota set for a directory



3. Analyze logs. The following log indicates that writing the file requires 384 MB space, but the current space quota is only 150 MB. Therefore, the space is insufficient. Before a file is written, the required remaining space is as follows: Block size x Number of copies. 128 MB x 3 copies = 384 MB.

```
[omm@189-39-150-115 client]$
[omm@189-39-150-115 client]$ hdfs dfs -put switchuser.py  /tmp/tquota2
put: The DiskSpace quota of /tmp/tquota2 is exceeded: quota = 157286400 B = 150 MB but diskspace consumed = 402653184 B = 384 MB
```

Solution

Step 1 Set a proper quota for the directory.

hadoop dfsadmin -setSpaceQuota 150G /*directory name*

Step 2 Run the following command to clear the quota:

hdfs dfsadmin -clrSpaceQuota /*directory name*

----End

16.9.22 Balancing Fails, and "Source and target differ in block-size" Is Displayed

Symptom

When the **distcp** command is executed to copy files across clusters, the message "Source and target differ in block-size." is displayed, indicating that some files fail to be copied. Use -pb to preserve block-sizes during copy. "

```
 Caused by: java.io.IOException: Check-sum mismatch between hdfs://10.180.144.7:25000/kylin/kylin_default_instance_prod/parquet/f2e72874-f01c-45ff-b219-207f3a5b3fcb/c769cd2d-575a-4459-837b-a19dd7b20c27/339114721280/0.parquettar and hdfs://10.180.180.194:25000/kylin/kylin_default_instance_prod/parquet/f2e72874-f01c-45ff-b219-207f3a5b3fcb/.distcp.tmp.attempt_1523424430246_0004_m_000019_2. Source and target differ in block-size. Use -pb to preserve block-sizes during copy. Alternatively, skip checksum-checks altogether, using -skipCrc. (NOTE: By skipping checksums, one runs the risk of masking data-corruption during file-transfer.)        at org.apache.hadoop.tools.mapred.RetriableFileCopyCommand.compareCheckSums(RetriableFileCopyCommand.java:214)
```

Possible Causes

This is not a version-related problem. When you run the **distcp** command to copy files, the block size of the source file is not recorded by default. As a result, the verification fails when the block size of the source file is not 128 MB. In this case, you need to add parameter **-pb** to the **distcp** command.

Cause Analysis

1. The block size is set when data is written to HDFS. The default block size is 128 MB. The size of files written by some components or service programs may not be 128 MB, for example, 8 MB.

```
<name>dfs.blocksize</name>
<value>134217728</value>
```

Figure 16-32 Size of files written by some components or service programs

2. DistCp reads the file from a source cluster and writes it to a destination cluster. By default, the value of dfs.blocksize in the MapReduce task is used as the block size, whose default value is 128 MB.
3. After DistCp finishes writing a file, the system performs verification based on the physical size of the block. Because the block size of the file in the source cluster is different from that of the file in the destination cluster, the splitting sizes are different. As a result, the verification fails.

   For example, in the preceding file, there are three blocks (17.9/8 MB = 3 blocks) in the old cluster and one block (17.9/128 MB = 1 block) in the new cluster. Therefore, the verification fails because the physical size of the disk is divided.

Solution

Add parameter **-pb** in the **distcp** command. This parameter is used to reserve the block size when **distcp** is used to ensure that the block size of the new cluster is the same as that of the old cluster.

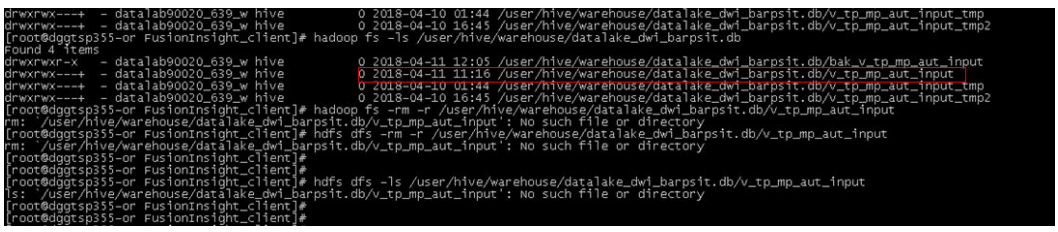
Figure 16-33 Size of the reserved block during **distcp** command execution

16.9.23 A File Fails to Be Queried or Deleted, and the File Can Be Viewed in the Parent Directory (Invisible Characters)

Symptom

A file fails to be queried or deleted using the HDFS Shell client. The file can be viewed in the parent directory.

Figure 16-34 List of files in the parent directory



Cause Analysis

The possible cause is that invisible characters are written to the file. You can write the file name to the local text and run the **vi** command to open the file.

hdfs dfs -ls *parent directory* **> /tmp/t.txt**

vi /tmp/t.txt

Run the **:set list** command to display invisible characters in the file name. For example, the file name contains **^M**, which is invisible.

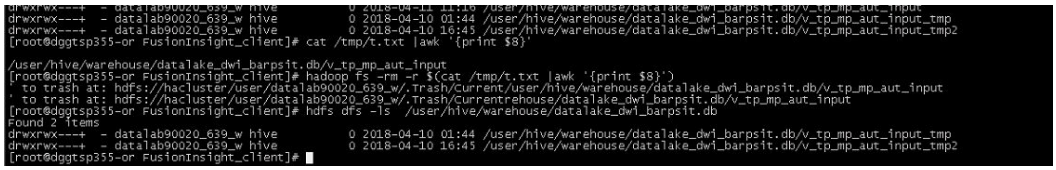
Figure 16-35 Displaying invisible characters

Solution

Step 1 Run the Shell command to read the file name recorded in the text. Ensure that the following command output contains the full path of the file in HDFS.

cat /tmp/t.txt |awk '{print $8}'

Figure 16-36 File path



Step 2 Run the following command to delete the file:

hdfs dfs -rm $(cat /tmp/t.txt |awk '{print $8}')

Step 3 Verify that the file has been deleted.

hdfs dfs -ls *parent directory*

----End

16.9.24 Uneven Data Distribution Due to Non-HDFS Data Residuals

Symptom

Data distribution is uneven. A disk is full while other disks have sufficient space.

The data storage directory of HDFS DataNode is set to **/export/data1/dfs--/export/data12/dfs**. A large volume of data is stored to **/export/data1/dfs** but data is evenly distributed to other disks.

Cause Analysis

The customer's disk is reinstalled. However, a directory is not thoroughly deleted during disk uninstallation, that is, the added disk is unformatted and historical junk data remains.

Solution

Manually delete data residuals.

16.9.25 Uneven Data Distribution Due to the Client Installation on the DataNode

Symptom

Data is unevenly distributed on HDFS DataNodes. Disk usage of a node is high or even reaches 100% while disks on other nodes have sufficient idle space.

Cause Analysis

In the HDFS data replica mechanism, the first replica is stored to the local node where the client is stored. As a result, disks of the node run out while disks of other nodes have sufficient idle space.

Solution

Step 1 For the existing data unevenly distributed, run the following command to balance data:

/opt/client/HDFS/hadoop/sbin/start-balancer.sh -threshold 10

/opt/client indicates the actual client installation directory.

Step 2 For new data, install the client on the node without DataNode.

----End

16.9.26 Handling Unbalanced DataNode Disk Usage on Nodes

Symptom

The disk usage of each DataNode on a node is uneven.

Example:

```
189-39-235-71:~ # df -h
Filesystem  Size  Used Avail Use% Mounted on
/dev/xvda  360G  92G   250G  28% /
/dev/xvdb  700G  900G   200G  78% /srv/BigData/hadoop/data1
/dev/xvdc  700G  900G   200G  78% /srv/BigData/hadoop/data2
/dev/xvdd  700G  900G   200G  78% /srv/BigData/hadoop/data3
/dev/xvde  700G  900G   200G  78% /srv/BigData/hadoop/data4
/dev/xvdf  10G   900G   890G  2% /srv/BigData/hadoop/data5
189-39-235-71:~ #
```

Possible Causes

Some disks are faulty and are replaced with new ones. The new disk usage is low.

Disks are added. For example, the original four data disks are expanded to five disks.

Cause Analysis

There are two policies for writing data to Block disks on DataNodes: 1. Round Robin (default value) and 2. Preferentially writing data to the disk with the more available space.

Description of the **dfs.datanode.fsdataset.volume.choosing.policy** parameter

Possible values:

- Polling: **org.apache.hadoop.hdfs.server.datanode.fsdataset.RoundRobinVolumeChoosingPolicy**
- Preferentially writing data to the disk with more available space: **org.apache.hadoop.hdfs.server.datanode.fsdataset.AvailableSpaceVolumeChoosingPolicy**

Solution

Change the value of **dfs.datanode.fsdataset.volume.choosing.policy** to **org.apache.hadoop.hdfs.server.datanode.fsdataset.AvailableSpaceVolumeChoosingPolicy**, save the settings, and restart the affected services or instances.

In this way, the DataNode preferentially selects a node with the most available disk space to store data copies.

NOTE

- Data written to the DataNode will be preferentially written to the disk with more available disk space.
- The high usage of some disks can be relieved with the gradual deletion of aging data from the HDFS.

16.9.27 Locating Common Balance Problems

Problem 1: Lack of Permission to Execute the balance Task (Access denied).

Problem details: After the **start-balancer.sh** command is executed, the " hadoop-root-balancer-hostname.out" log displays "Access denied for user test1. Superuser privilege is required."

```
cat /opt/client/HDFS/hadoop/logs/hadoop-root-balancer-host2.out
Time Stamp               Iteration#  Bytes Already Moved  Bytes Left To Move  Bytes Being Moved
INFO: Watching file:/opt/client/HDFS/hadoop/etc/hadoop/log4j.properties for changes with interval : 60000
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.AccessControlException): Access denied for user test1.
Superuser privilege is required
at
org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkSuperuserPrivilege(FSPermissionChecker.java:122)
at
org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkSuperuserPrivilege(FSNamesystem.java:5916)
```

Cause analysis:

The administrator account is required for executing the balance task.

Solution

- Secure version

  Perform authentication for user **hdfs** or a user in the **supergroup** group and then execute the balance task.
- General version

Run the **su - hdfs** command on the client before running the **balance** command on HDFS.

Problem 2: The balance command fails to be executed, and the /system/balancer.id file is abnormal.

Problem details:

A user starts a balance process on the HDFS client. After the process is stopped unexpectedly, the user performs the balance operation again. The operation fails.

```
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.protocol.RecoveryInProgressException): Failed to APPEND_FILE /system/balancer.id for DFSClient because lease recovery is in progress. Try again later.
```

Cause analysis:

Generally, after the balance operation is complete in HDFS, the **/system/balancer.id** file is automatically released and the balance operation can be performed again.

In the preceding scenario, the first balance operation is stopped abnormally. Therefore, when the balance operation is performed for the second time, the **/system/balancer.id** file still exists. As a result, the **append /system/balancer.id** operation is triggered and the balance operation fails.

Solution

Method 1: After the hard lease period exceeds one hour, release the lease on the original client and perform the balance operation again.

Method 2: Delete the **/system/balancer.id** file from HDFS and perform the balance operation again.

16.9.28 HDFS Displays Insufficient Disk Space But 10% Disk Space Remains

Symptom

1. The alarm "HDFS Disk Usage Exceeds the Threshold" is reported.
2. On the HDFS page, high disk space usage is displayed.

Cause Analysis

The **dfs.datanode.du.reserved.percentage** parameter is set in HDFS, indicating the percentage of the reserved space of each disk to the total disk space. The DataNode reserves space you set for NodeManager running and computing of other components, for example, Yarn, or for upgrades.

As 10% disk space is reserved, the HDFS DataNode regards that there is no available disk space when the disk usage reaches 90%.

Solution

Step 1 Expand the HDFS DataNode disk capacity when its usage reaches 80%.

Step 2 If the disk capacity cannot be expanded in time, delete useless data in HDFS to release disk space.

----End

16.9.29 An Error Is Reported When the HDFS Client Is Installed on the Core Node in a Common Cluster

Issue

In a common cluster, an error message is displayed when a user is created on the Core node to install the client.

Symptom

In a common cluster, the following error message is displayed when a user is created on the Core node to install the client:

```
2020-03-14 19:16:17,166 WARN shortcircuit.DomainSocketFactory: error creating DomainSocket
java.net.ConnectException: connect(2) error: Permission denied when trying to connect to '/var/run/MRS-HDFS/dn_socket'
at org.apache.hadoop.net.unix.DomainSocket.connect0(Native Method)
at org.apache.hadoop.net.unix.DomainSocket.connect(DomainSocket.java:256)
at org.apache.hadoop.hdfs.shortcircuit.DomainSocketFactory.createSocket(DomainSocketFactory.java:168)
at org.apache.hadoop.hdfs.client.impl.BlockReaderFactory.nextDomainPeer(BlockReaderFactory.java:799)
...
```

Cause Analysis

A user runs the **useradd** command to create a user. The default user group of the user does not contain the **ficommmon** user group. As a result, the preceding error is reported when the **get** command of HDFS is executed.

Procedure

Run the **usermod -a -G ficommon username** command to add the user to the **ficommon** user group.

16.9.30 Client Installed on a Node Outside the Cluster Fails to Upload Files Using hdfs

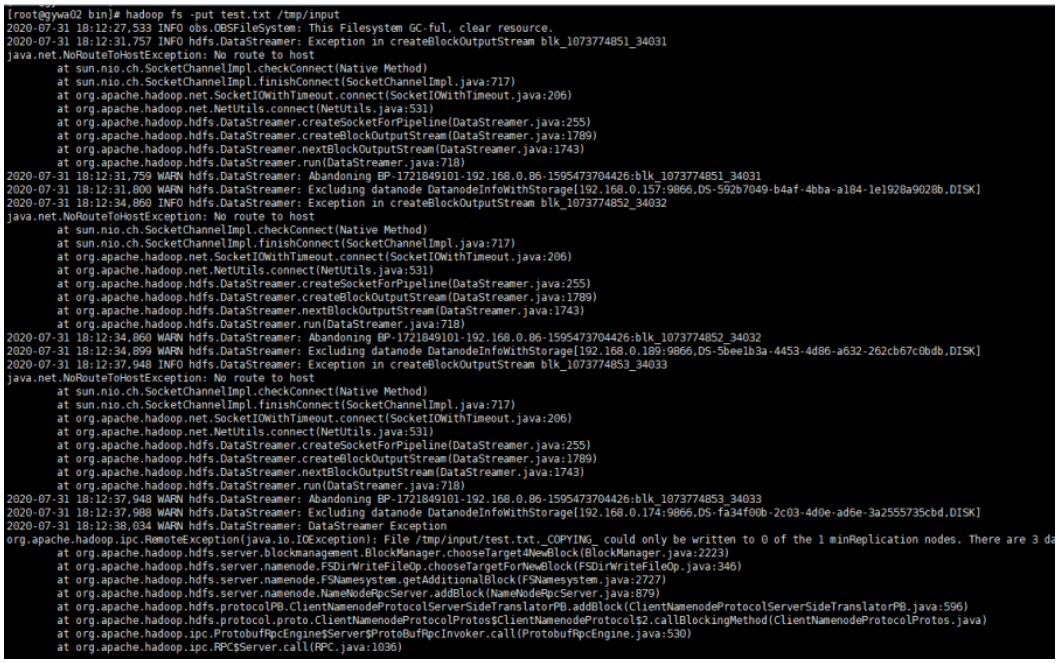
Issue

A client installed on a node outside the cluster fails to upload files using hdfs.

Symptom

After a client is installed on a cluster node and a file is uploaded using the **hdfs** command, the following error is reported:

Figure 16-37 An error is reported during file upload.



Cause Analysis

The error message "no route to host" is displayed, and the IP address 192.168 is contained in the error message. That is, the internal network route from the client node to the DataNode in the cluster is unreachable. As a result, the file fails to be uploaded.

Procedure

In the client directory of the client node, find the **hdfs-site.xml** file in the HDFS client configuration directory. Add the **dfs.client.use.datanode.hostname** configuration item to the configuration file, and set the value to **true**.

16.9.31 Insufficient Number of Replicas Is Reported During High Concurrent HDFS Writes

Symptom

File writes to HDFS fail occasionally.

The operation log is as follows:

105 | INFO | IPC Server handler 23 on 25000 | IPC Server handler 23 on 25000, call org.apache.hadoop.hdfs.protocol.ClientProtocol.addBlock from 192.168.1.96:47728 Call#1461167 Retry#0 | Server.java:2278
java.io.IOException: File /hive/warehouse/000000_0.835bf64f-4103 could only be replicated to 0 nodes instead of minReplication (=1). **There are 3 datanode(s) running and 3 node(s) are excluded in this operation.**

Cause Analysis

- HDFS has a reservation mechanism for file writing: each block to be written is 128 MB no matter whether the file is 10 MB or 1 GB. If a 10 MB file needs to

be written, the file occupies 10 MB of the first block and about 118 MB space will be released. If a 1 GB file needs to be written, HDFS writes the file block by block and releases unused space after the file is written.

- If there are a large number of files to be written concurrently, the disk space for reserved write blocks is insufficient. As a result, the file fails to be written.

Solution

Step 1 Log in to the HDFS WebUI and go to the JMX page of the DataNode.

1. On the native HDFS page, choose **Datanodes**.
2. Locate the target DataNode and click the HTTP address to go to the DataNode details page.
3. Change **datanode.html** in **url** to **jmx**.

Step 2 Search for the **XceiverCount** indicator. If the value of this indicator multiplied by the block size exceeds the DataNode disk capacity, the disk space reserved for block write is insufficient.

Step 3 You can use either of the following methods to solve the problem:

Method 1: Reduce the service concurrency.

Method 2: Combine multiple files into one file to reduce the number of files to be written.

----End

16.9.32 HDFS Client Failed to Delete Overlong Directories

Symptom

When a user runs the **hadoop fs -rm -r -f obs://***<obs_path>* command to delete an OBS directory with an overlong path name, the following error message is displayed:

```
2022-02-28 17:12:45,605 INFO internal.RestStorageService: OkHttp cost 19 ms to apply http request
2022-02-28 17:12:45,606 WARN internal.RestStorageService: Request failed, Response code: 400; Request ID: 0000017F3F9A8545401491602FC8CAD9; Request path: http://wordcount01-fcq.obs.xxx.***.example.com/user%2Froot%2F.Trash%2FCurrent
%2Ftest1%2F12345678901234567890123456789012345678901234567890123456789012345678901234567
8901234567890123456789012345678901234567890123456789012345678901234567890123456789012345
6789012345678901234567890123456789012345678901234567890123456789012345678901234567890123
4567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
2345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567
8901234567890123456789012345678901234567890123456789012345678901234567890123456789012345
6789012345678901234567890123456789012345678901234567890123456789012345678901234567890123
4567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
2345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567
8901234567890123456789012345678901234567890123456789012345678901234567890123456
2022-02-28 17:12:45,606 WARN services.AbstractClient: Storage|1|HTTP+XML|getObjectMetadata|||| 2022-02-28 17:12:45|2022-02-28 17:12:45|||400|
2022-02-28 17:12:45,607 INFO log.AccessLogger: 2022-02-28 17:12:45 605| com.obs.services.internal.RestStorageService|executeRequest|560|OkHttp cost 19 ms to apply http request
2022-02-28 17:12:45 606|com.obs.services.internal.RestStorageService|handleThrowable|221|Request failed, Response code: 400; Request ID: 0000017F3F9A8545401491602FC8CAD9; Request path: http://wordcount01-fcq.obs.xxx.***.example.com/user%2Froot%2F.Trash%2FCurrent
%2Ftest1%2F12345678901234567890123456789012345678901234567890123456789012345678901234567
```

```
8901234567890123456789012345678901234567890123456789012345678901234567890123456789012345
6789012345678901234567890123456789012345678901234567890123456789012345678901234567890123
4567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
2345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567
8901234567890123456789012345678901234567890123456789012345678901234567890123456789012345
6789012345678901234567890123456789012345678901234567890123456789012345678901234567890123
4567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
2345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567
8901234567890123456789012345678901234567890123456
2022-02-28 17:12:45 606|com.obs.services.AbstractClient|doActionWithResult|404|Storage|1|HTTP+XML|
getObjectMetadata||||2022-02-28 17:12:45|2022-02-28 17:12:45|||400|
```

Cause Analysis

When you run the **rm** command to delete some content from the HDFS, the files or directories are not deleted immediately. Instead, they are moved to the **current** directory (**/user/*${Username}*/.Trash/current**) in the recycle bin.

Solution

You can run the **skipTrash** command to skip the HDFS recycle bin and directly delete the data. Set the **dfs.client.skipTrash.enabled=true** configuration item of the HDFS client.

Step 1 Log in to any master node in the cluster as user **root**.

Step 2 Run the following command to edit the **hdfs-site.xml** file used by HDFS:

vim *Client installation directory*/**HDFS/hadoop/etc/hadoop/hdfs-site.xml**

Step 3 Add the following content to the **hdfs-site.xml** file:

```
<property>
<name>dfs.client.skipTrash.enabled</name>
<value>true</value>
</property>
```

Step 4 Run the following command to delete the overlong OBS directory:

hadoop fs -rm -r -f –skipTrash obs://*<obs_path>*

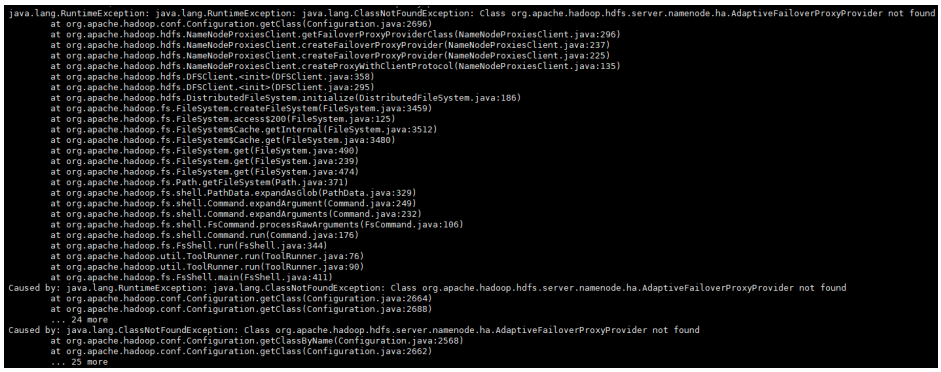
Step 5 Log in to the other master nodes in the cluster and repeat **Step 2** to **Step 4** one by one until the operations are complete on all master nodes.

----End

16.9.33 An Error Is Reported When a Node Outside the Cluster Accesses MRS HDFS

Symptom

When a node outside the MRS cluster accesses HDFS in the cluster, the following error message is reported: Class org.apache.hadoop.hdfs.server.namenode.ha.AdaptiveFailoverProxyProvider not found



Cause Analysis

The possible causes are as follows:

- An error is reported when an open-source HDFS client accesses HDFS of an MRS cluster.
- An error is reported when the JAR package is used to connect to HDFS of the MRS cluster (including connection to HDFS during task submission).

Procedure

Method 1:

Step 1 Locate the HDFS configuration file **hdfs-site.xml** used by the command or JAR package.

Step 2 Modify the **dfs.client.failover.proxy.provider.hacluster** configuration as follows:

```
<property>
<name>dfs.client.failover.proxy.provider.hacluster</name>
<value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider</value>
</property>
```

NOTE

You can also delete the preceding configuration items.

Step 3 Save the file and access MRS HDFS again.

----End

Method 2:

Step 1 Download the hadoop-plugins matching the MRS cluster version from the Maven repository.

Step 2 Add the downloaded JAR package to the dependency of the command or JAR package.

----End

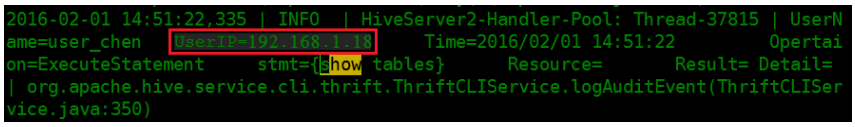
16.10 Using Hive

16.10.1 Content Recorded in Hive Logs

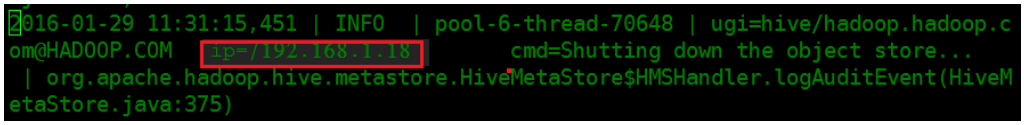
Audit log

An audit log records at what time a user sends a request to HiveServer and MetaStore from which IP address with what statement.

The following HiveServer audit log shows that at 14:51:22 on February 1, 2016, **user_chen** sent a **show tables** request to HiveServer from the 192.168.1.18 IP address.



The following MetaStore audit log shows that user **hive** sent a **shutdown** request to MetaStore from the 192.168.1.18 IP address at 11:31:15 on January 29, 2016.



Generally, the audit log does not play a role in actual error location. However, the audit log must be checked to solve the following problems:

1. There is no response after a client sends a request. The audit log can be used to check whether the task suspends on the client or server. If the audit log has no related information, the task suspends on the client. If the audit log has related information, view the run log to locate where the program suspends.
2. The audit log can be used to check the number of requests in a specified period of time. You can view the number of requests in a specified period in audit logs.

HiveServer Run Log

HiveServer receives requests from a client (SQL statement), compile and execute the statement (submitted to Yarn or local MapReduce), and interact with MetaStore to obtain metadata information. The HiveServer run log records a complete SQL execution process.

Generally, if SQL statement running fails, check the HiveServer run log first.

MetaStore Run Log

Typically, if the HiveServer run log contains MetaException or MetaStore connection failure, check the MetaStore run log.

GC Log

Both HiveServer and MetaStore have GC logs. If GC-related problems occur, view the GC logs to quickly locate the cause. For example, if HiveServer or MetaStore frequently restarts, check its GC log.

16.10.2 Causes of Hive Startup Failure

The most common cause of the Hive startup failure is that the MetaStore instance cannot connect to DBService. You can view the detailed error information in the MetaStore logs. The reasons for the failure to connect to DBService are as follows:

Possible Cause 1

DBService does not properly initialize the Hive metabase hivemeta.

Procedure 1

Step 1 Run the following commands:

source /opt/Bigdata/MRS_*XXX*/install/dbservice/.dbservice_profile

gsql -h *DBservice floating IP* **-p 20051 -d hivemeta -U hive -W HiveUser@**

Step 2 If the interaction interface cannot be properly displayed, database initialization fails. If the following error information is displayed, the hivemeta configuration may be lost in the configuration file of the node where DBService is located.

```
org.postgresql.util.PSQLException: FATAL: no pg_hba.conf entry for host "192.168.0.146", database "HIVEMETA"
```

Step 3 Edit **/srv/BigData/dbdata_service/data/pg_hba.conf** by adding **host hivemeta hive 0.0.0.0/0 sha256** to the file.

Step 4 Run the **source /opt/Bigdata/MRS_*XXX*/install/dbservice/.dbservice_profile** command to configure environment variables.

Step 5 Run **gs_ctl -D $GAUSSDATA reload #** to make new configurations take effect.

----End

Possible Cause 2

The floating IP address of DBService is incorrect. As a result, the IP address of the MetaStore node fails to connect to or build mutual trust with the floating IP address, causing MetaStore startup failure.

Procedure 2

The floating IP address of DBService must be an IP address that is not used in the same network segment and cannot be pinged before configuration. Modify the floating IP address of DBService.

16.10.3 "Cannot modify *xxx* at runtime" Is Reported When the set Command Is Executed in a Security Cluster

Symptom

The following error is reported when running the **set** command:

```
0: jdbc:hive2://192.168.1.18:21066/> set mapred.job.queue.name=QueueA;
 Error: Error while processing statement: Cannot modify mapred.job.queue.name at list of params that are allowed to be modified at runtime (state=42000,code=1)
```

Procedure

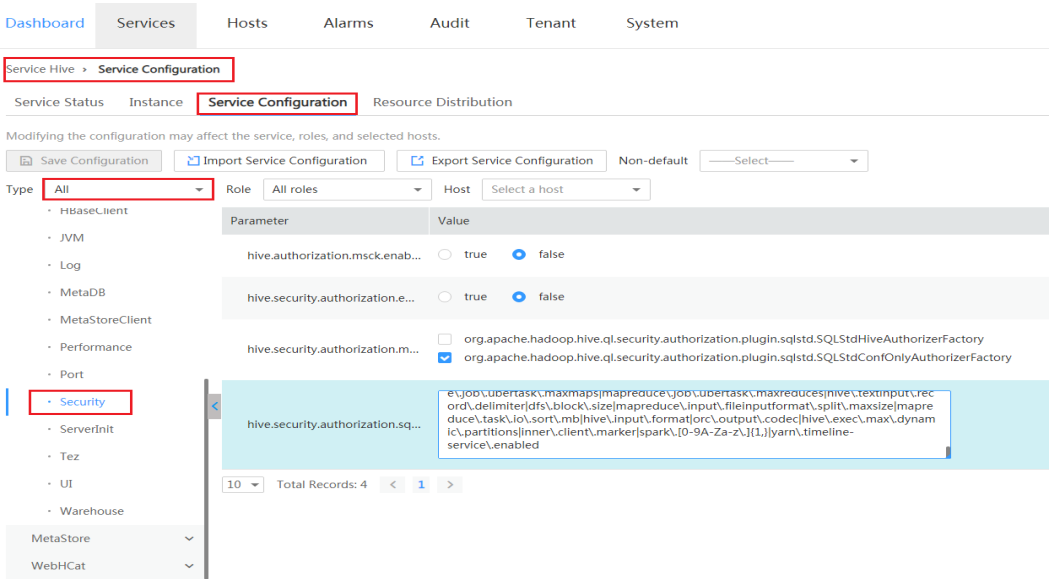
Solution 1:

Step 1 Log in to Manager and modify Hive parameters.

- MRS Manager: Log in to MRS Manager and choose **Services** > **Hive** > **Service Configuration**. Set **Type** to **All** and choose **HiveServer** > **Security**.
- FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Services** > **Hive** > **Configurations** > **All Configurations** > **HiveServer** > **Security**.

Step 2 Add the command parameters to be executed to the **hive.security.authorization.sqlstd.confwhitelist.append** configuration item.

Step 3 Click **Save** and restart **HiveServer**.



----End

Solution 2:

Step 1 Log in to Manager and modify Hive parameters.

- MRS Manager: Log in to MRS Manager and choose **Services** > **Hive** > **Service Configuration**. Set **Type** to **All** and choose **HiveServer** > **Security**.
- FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Services** > **Hive** > **Configurations** > **All Configurations** > **HiveServer** > **Security**.

Step 2 Locate **hive.security.whitelist.switch** and select **OFF**. Click **Save** and restart HiveServer.

----End

16.10.4 How to Specify a Queue When Hive Submits a Job

Symptom

How do I specify a queue when Hive submits a job?

Procedure

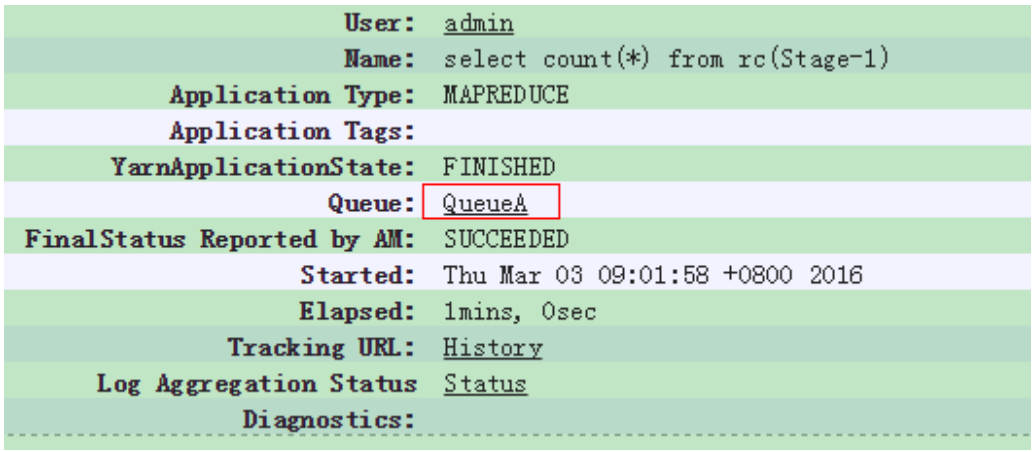
Step 1 Before submitting the job, set the job queue, for example, submitting the job to QueueA.

```
set mapred.job.queue.name=QueueA;
select count(*) from rc;
```

NOTE

The queue name is case sensitive. For example, in this example, **queueA** and **Queuea** are invalid. In addition, the queue must be a leaf queue, and jobs cannot be submitted to a non-leaf queue.

Step 2 After job submission, go to the Yarn page to check the job. The job has been submitted to QueueA.



----End

16.10.5 How to Set Map and Reduce Memory on the Client

Symptom

How do I set Map and Reduce memory on the client?

Procedure

Before SQL statement execution, run the set command to set parameters of clients related to Map/Reduce.

The following parameters are related to Map and Reduce memory:

```
set mapreduce.map.memory.mb=4096; //Memory required by each Map task
set mapreduce.map.java.opts=-Xmx3276M; //Maximum memory used by the JVM of each Map task
set mapreduce.reduce.memory.mb=4096; //Memory required by each Reduce task
set mapreduce.reduce.java.opts=-Xmx3276M; //Maximum memory used by the JVM of each Reduce task
set mapred.child.java.opts=-Xms1024M -Xmx3584M; // This parameter is a global parameter, which is used to set Map and Reduce in a unified manner.
```

NOTE

Parameter settings take effect for the current session only.

16.10.6 Specifying the Output File Compression Format When Importing a Table

Question

How do I specify an output file compression format when importing a table?

Procedure

Hive supports the following compression formats:

```
org.apache.hadoop.io.compress.BZip2Codec
org.apache.hadoop.io.compress.Lz4Codec
org.apache.hadoop.io.compress.DeflateCodec
org.apache.hadoop.io.compress.SnappyCodec
org.apache.hadoop.io.compress.GzipCodec
```

- If global settings are required, that is, all tables need to be compressed, you can perform the following global settings for Hive service configuration parameters on the Manager page:
  - Set **hive.exec.compress.output** to **true**.
  - Set **mapreduce.output.fileoutputformat.compress.codec** to **org.apache.hadoop.io.compress.BZip2Codec**.

    NOTE

    The following parameters take effect only when **hive.exec.compress.output** is set to **true**.
- If it needs to be set at the session level, configure the parameters as follows before command execution:

```
set hive.exec.compress.output=true;
set mapreduce.output.fileoutputformat.compress.codec=org.apache.hadoop.io.compress.SnappyCodec;
```

16.10.7 desc Table Cannot Be Completely Displayed

Symptom

How do I make sure that the description is completely displayed when the desc table is too long?

Procedure

Step 1 When starting Beeline of Hive, set **maxWidth** to **20000**.

```
[root@192-168-1-18 logs]# beeline --maxWidth=20000
scan complete in 3ms
Connecting to
...
Beeline version 1.1.0 by Apache Hive
```

Step 2 (Optional) Run the **beeline -help** command to view the client display settings.

```
 -u <database url>               the JDBC URL to connect to
 -n <username>                   the username to connect as
 -p <password>                   the password to connect as
 -d <driver class>               the driver class to use
 -i <init file>                  script file for initialization
 -e <query>                      query that should be executed
 -f <exec file>                  script file that should be executed
```

```
 --hiveconf property=value       Use value for given property
 --color=[true/false]             control whether color is used for display
 --showHeader=[true/false]        show column names in query results
 --headerInterval=ROWS;           the interval between which heades are displayed
 --fastConnect=[true/false]       skip building table/column list for tab-completion
 --autoCommit=[true/false]        enable/disable automatic transaction commit
 --verbose=[true/false]           show verbose error messages and debug info
 --showWarnings=[true/false]      display connection warnings
 --showNestedErrs=[true/false]    display nested errors
 --numberFormat=[pattern]         format numbers using DecimalFormat pattern
 --force=[true/false]             continue running script even after errors
 --maxWidth=MAXWIDTH              the maximum width of the terminal
 --maxColumnWidth=MAXCOLWIDTH     the maximum width to use when displaying columns
 --silent=[true/false]            be more silent
 --autosave=[true/false]          automatically save preferences
 --outputformat=[table/vertical/csv2/tsv2/dsv/csv/tsv]  format mode for result display
                                Note that csv, and tsv are deprecated - use csv2, tsv2 instead
 --truncateTable=[true/false]     truncate table column when it exceeds length
 --delimiterForDSV=DELIMITER      specify the delimiter for delimiter-separated values output format
(default: |)
 --isolation=LEVEL                set the transaction isolation level
 --nullemptystring=[true/false]  set to true to get historic behavior of printing null as empty string
 --socketTimeOut=n                socket connection timeout interval, in second. The default value is 300.
```

----End

16.10.8 NULL Is Displayed When Data Is Inserted After the Partition Column Is Added

Symptom

1. Run the following command to create a table:
   ```
   create table test_table(
   col1 string,
   col2 string
   )
   PARTITIONED BY(p1 string)
   STORED AS orc tblproperties('orc.compress'='SNAPPY');
   ```
2. Modify the table structure, add partitions, and insert data.
   ```
   alter table test_table add partition(p1='a');
   insert into test_table partition(p1='a') select col1,col2 from temp_table;
   ```
3. Modify the table structure, add columns, and insert data.
   ```
   alter table test_table add columns(col3 string);
   insert into test_table partition(p1='a') select col1,col2,col3 from temp_table;
   ```
4. Query data in the **test_table** table. In the returned result, the values in the **col3** column are all NULL.
   ```
   select * from test_table where p1='a'
   ```
5. Add a table partition and insert data.
   ```
   alter table test_table add partition(p1='b');
   insert into test_table partition(p1='b') select col1,col2,col3 from temp_table;
   ```
6. Query data in the **test_table** table. In the returned result, the value of **col3** is not all NULL.
   ```
   select * from test_table where p1='b'
   ```

Cause Analysis

RESTRICT is the default option for altering a table. In the RESTRICT mode, only the metadata is changed, while the table's partition structure created before the altering operation remains unchanged. However, new partitions created after the altering operation are changed. Therefore, when values of the old partitions are queried, they are all NULL.

Procedure

Add the **cascade** keyword when adding columns, for example:

```
alter table test_table add columns(col3 string) cascade;
```

16.10.9 A Newly Created User Has No Query Permissions

Symptom

When a user is created, an error message is displayed indicating that the user does not have permissions to query data.

```
Error: Error while compiling statement: FAILED: HiveAccessControlException Permission denied: Principal [name=hive, type=USER] does not have following privileges for operation QUERY [[SELECT] on Object [type=TABLE_OR_VIEW, name=default.t1]] (state=42000,code=40000)
```

Cause Analysis

The newly created user does not have the permission to operate the Hive component.

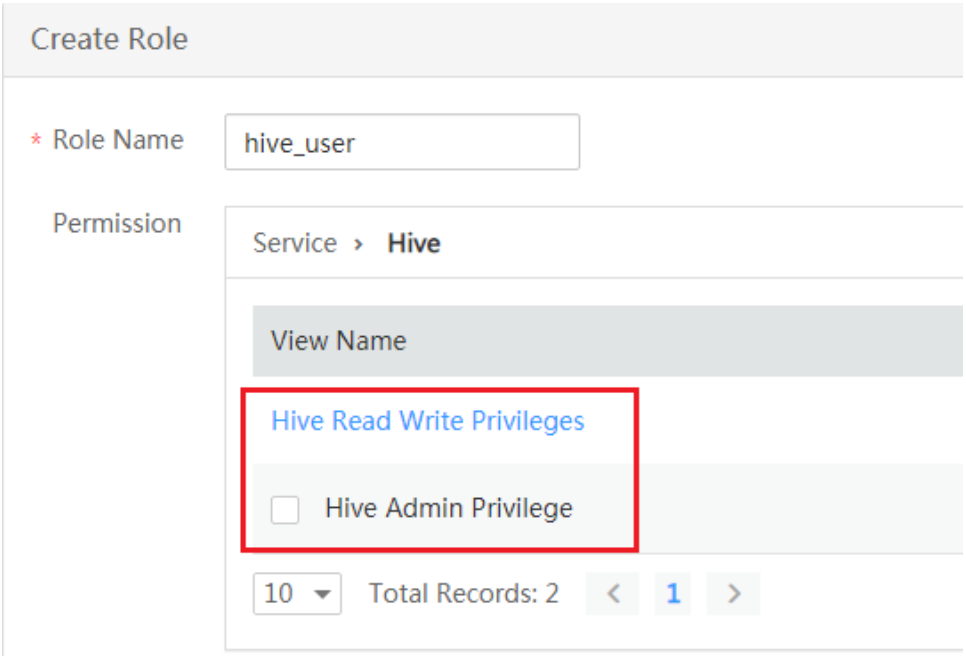
Solution

MRS Manager:

Step 1 Log in to MRS Manager and choose **System** > **Manage Role** > **Create Role**.

Step 2 Enter a role name.

Step 3 In the **Permission** area, select **Hive**. The Hive administrator permission and the read and write permission for Hive tables are displayed.



Step 4 Select **Hive Read Write Privileges**. All databases in the Hive column are displayed.

Step 5 Select the permissions required by the role and click **OK**.

Step 6 On MRS Manager, choose **System** > **Manage User**.

Step 7 Locate the row that contains the created user, and click **Modify** in the **Operation** column.

Step 8 Click **Select and Join User Group**. To use the Hive service, you must add a Hive group.

Step 9 Click **Select and Add Role** and select the role created in **Step 5**.

Step 10 Click **OK**.

----End

FusionInsight Manager:

Step 1 Log in to FusionInsight Manager. Choose **System** > **Permission** > **Role**.

Step 2 Click **Create Role**, and set **Role name** and **Description**.

Step 3 Set **Configure Resource Permission** for the role and select **Hive Read and Write Permission** for the Hive table. All databases in the Hive column are displayed.

Step 4 Select the permissions required by the role and click **OK**.

Step 5 On FusionInsight Manager, choose **System** > **Permission** > **User**.

Step 6 Locate the row that contains the created user, and click **Modify** in the **Operation** column.

Step 7 Click **Add** on the right of **User Group**. To use the Hive service, you must add a Hive group.

Step 8 Click **Add** on the right of **Role** and select the role created in **4**.

Step 9 Click **OK**.

----End

16.10.10 An Error Is Reported When SQL Is Executed to Submit a Task to a Specified Queue

Symptom

The following error is reported when executing SQL to submit a task to Yarn:

```
Failed to submit application_1475400939788_0033 to YARN :
org.apache.hadoop.security.AccessControlException: User newtest cannot submit applications to queue
root.QueueA
```

Cause Analysis

The current login user does not have the permission to submit the YARN queue.

Solution

Grant the submission permission of the specified Yarn queue to the user. On Manager, choose **System** > **Permission** > **User** and bind a role with the queue submission permission to the user.

16.10.11 An Error Is Reported When the "load data inpath" Command Is Executed

Symptom

The following errors are reported when the **load data inpath** command is executed:

- Error 1:
  ```
  HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does not have following privileges on Object [type=DFS_URI, name=hdfs://hacluster/tmp/input/mapdata] for operation LOAD : [OBJECT OWNERSHIP]
  ```
- Error 2:
  ```
  HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does not have following privileges on Object [type=DFS_URI, name=hdfs://hacluster/tmp/input/mapdata] for operation LOAD : [INSERT, DELETE]
  ```
- Error 3:
  ```
  SemanticException [Error 10028]: Line 1:17 Path is not legal ''file:///tmp/input/mapdata'': Move from: file:/tmp/input/mapdata to: hdfs://hacluster/user/hive/warehouse/tmp1 is not valid. Please check that values for params "default.fs.name" and "hive.metastore.warehouse.dir" do not conflict.
  ```

Cause Analysis

The current login user does not have the permission to operate the directory or the file directory format is incorrect.

Solution

Hive has the following requirements on the **load data inpath** command:

- The file owner must be the user who executes the command.
- The current user must have read and write permissions for the file.
- The current user must have permissions to execute the directory of the file.
- The current user must have the write permission on the directory of the table, because the load operation moves the file to the directory.
- The file format must be the same as the storage format specified by the table. For example, if **stored as rcfile** is specified during table creation but the file format is TXT, it is unsatisfied.
- The file must be stored in HDFS. Files in the local file system cannot be specified using the **file://** form.
- The file name cannot start with an underscore (_) or period (.). A file whose name starts with an underscore (_) or period (.) will be ignored.

  The following shows permissions required when user **test_hive** loads data.

  ```
  [root@192-168-1-18 duan]# hdfs dfs -ls /tmp/input2
  16/03/21 14:45:07 INFO hdfs.PeerCache: SocketCache disabled.
  Found 1 items
  -rw-r--r--   3 test_hive hive          6 2016-03-21 14:44 /tmp/input2/input.txt
  ```

16.10.12 An Error Is Reported When the "load data local inpath" Command Is Executed

Symptom

The following errors are reported when the **load data local inpath** command is executed:

- Error 1:
  ```
  HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does not have following privileges on Object [type=LOCAL_URI, name=file:/tmp/input/mapdata] for operation LOAD : [SELECT, INSERT, DELETE]
  ```
- Error 2:
  ```
  HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does not have following privileges on Object [type=LOCAL_URI, name=file:/tmp/input/mapdata] for operation LOAD : [OBJECT OWNERSHIP]
  ```
- Error 3:
  ```
  SemanticException Line 1:23 Invalid path ''/tmp/input/mapdata'': No files matching path file:/tmp/input/mapdata
  ```

Cause Analysis

The current user does not have the permission to operate the directory or the directory does not exist on the node where HiveServer is located.

Solution

NOTE

Generally, you are not advised to use local files to load data to Hive tables. You are advised to store local files in HDFS and then load data from the cluster.

Hive has the following requirements on the **load data local inpath** command:

- The file must be stored on the HiveServer node, because all commands are sent to the active HiveServer for execution.
- User **omm** must have the read permission for the file and read and execution permissions for the directory where the file is located, because the HiveServer process is started by user **omm** in the OS.
- The file owner must be the user who executes the command.
- The current user must have read and write permissions for the file.
- The file format must be the same as the storage format specified by the table. For example, if **stored as rcfile** is specified during table creation but the file format is TXT, it is unsatisfied.
- The file name cannot start with an underscore (_) or period (.). A file whose name starts with an underscore (_) or period (.) will be ignored.

16.10.13 An Error Is Reported When the "create external table" Command Is Executed

Symptom

The following error is reported when the **create external table** *xx(xx int)* **stored as textfile location '/tmp/aaa/aaa'** command is executed.

```
Permission denied. Principal [name=fantasy, type=USER] does not have following privileges on Object [type=DFS_URI, name=/tmp/aaa/aaa] for operation CREATETABLE : [SELECT, INSERT, DELETE, OBJECT OWNERSHIP] (state=42000,code=40000)
```

Cause Analysis

The current login user does not have the read and write permissions for the directory or its parent directory. When an external table is created, whether the current user is checked for its read and write permissions for the specified directory and its subdirectories and subfiles. If the specified directory does not exist, permissions for the parent directory are checked, and so on. If the check results show that the user has no permissions on any directory, "insufficient permission" is reported instead of "The specified directory does not exist".

Solution

Check whether the current user has read and write permissions for the **/tmp/aaa/aaa** path. If the path does not exist, check whether the user has read and write permissions for its parent directory.

16.10.14 An Error Is Reported When the dfs -put Command Is Executed on the Beeline Client

Symptom

Run the following command:

dfs -put /opt/kv1.txt /tmp/kv1.txt

The following error is reported:

```
Permission denied. Principal [name=admin, type=USER] does not have following privileges onObject[type=COMMAND_PARAMS,name=[-put, /opt/kv1.txt, /tmp/kv1.txt]] for operation DFS : [ADMIN PRIVILEGE] (state=,code=1)
```

Cause Analysis

The current login user does not have the permissions to run the command.

Solution

If the current user has the **admin** role, run the **set role admin** command to switch to the **admin** role. If the user does not have the admin role, bind the user with the permissions of the corresponding role on the Manager page.

16.10.15 Insufficient Permissions to Execute the set role admin Command

Symptom

When a user runs the following command:

set role admin

The following error is reported:

```
0: jdbc:hive2://192.168.42.26:21066/> set role admin;
Error: Error while processing statement: FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.DDLTask. dmp_B doesn't belong to role admin (state=08S01,code=1)
```

Cause Analysis

The current user does not have the permissions of the **admin** role of Hive.

Solution

Step 1 Log in to Manager.

- For versions earlier than MRS 3.x, go to **Step 7**.
- For MRS 3.x or later, choose **Cluster** > **Services** > **Hive**. In the upper right corner of the **Dashboard** page, click **More** and check whether **Enable Ranger** is unavailable.
  - If yes, go to **Step 2**.
  - If no, go to **Step 7**.

Step 2 Choose **Cluster** > **Services** > **Ranger** and click **RangerAdmin** in the **Basic Information** area. The Ranger web UI is displayed.

Step 3 Click the username in the upper right corner, select **Log Out** to log out of the system, and log in to the system as user **rangeradmin**.

Step 4 On the homepage, click **Settings** and choose **Roles**.

Step 5 Click the role with **Role Name** set to **admin**. In the **Users** area, click **Select User** and select a username.

Step 6 Click **Add Users**, select **Is Role Admin** in the row where the username is located, and click **Save**.

Step 7 Choose **System** > **Permission** > **Role** and add a role with the Hive administrator permission.

Step 8 On FusionInsight Manager, choose **System** > **Permission** > **User**.

Step 9 In the **Operation** column of the user, click **Modify**.

Step 10 Bind a role that has the Hive administrator permissions to the user and click **OK**.

----End

16.10.16 An Error Is Reported When UDF Is Created Using Beeline

Symptom

Run the following command:

create function fn_test3 as 'test.MyUDF' using jar 'hdfs:///tmp/udf2/MyUDF.jar'

The following error is reported:

```
Error: Error while compiling statement: FAILED: HiveAccessControlException Permission denied: Principal [name=admin, type=USER] does not have following privileges for operation CREATEFUNCTION [[ADMIN PRIVILEGE] on Object [type=DATABASE, name=default], [ADMIN PRIVILEGE] on Object [type=FUNCTION, name=default.fn_test3]] (state=42000,code=40000)
```

Cause Analysis

To create a permanent function in Hive, role **admin** is required.

Solution

Run the **set role admin** command before running the statement.

16.10.17 Difference Between Hive Service Health Status and Hive Instance Health Status

Question

What is the difference between Hive service health status and Hive instance health status?

Solution

The Hive service health status is displayed on the **Services** page and has four values: **Good**, **Bad**, **Partially Healthy**, and **Unknown**. It depends not only on Hive service availability but also the service status of other related components. Simple SQL is used to check Hive service availability.

Hive instances consist of HiveServer and MetaStore. Their health status is determined by communications between instances and JMX and can be **Good** (normal communications), **Concerning** (abnormal communications), or **Unknown** (no communications).

16.10.18 Hive Alarms and Triggering Conditions

Hive Alarms

Alarm ID	Alarm Severity	Auto Clear	Alarm Name	Alarm Type
16000	Minor	TRUE	Percentage of Sessions Connected to the HiveServer to Maximum Number Allowed Exceeds the Threshold	Fault alarm
16001	Minor	TRUE	Hive Warehouse Space Usage Exceeds the Threshold	Fault alarm
16002	Minor	TRUE	The Successful Hive SQL Operations Lower than The Threshold	Fault alarm
16004	Critical	TRUE	Hive Service Unavailable	Fault alarm

Alarm Triggering Scenarios

- 16000: An alarm is triggered when the ratio of the number of sessions connected to HiveServer to the allowed total number of sessions exceeds the threshold. For example, if the number of connected sessions is 9, the allowed total number of sessions is 12, and the threshold is 70%, an alarm is triggered, because 9/12 > 70%.
- 16001: An alarm is triggered when the ratio of HDFS capacities used by Hive to total HDFS capacities allocated to Hive exceeds the threshold. For example, if 500 GB is allocated to Hive, Hive uses 400 GB, and the threshold is 75%, an alarm is triggered, because 400/500 > 75%.
- 16002: An alarm is triggered when SQL execution success rate is lower than the threshold. If two out of four SQL statements are executed successfully and the threshold is 60%, an alarm is triggered, because 2/4 < 60%.
- 16004: An alarm is triggered when the health status of the Hive service changes to Bad.

NOTE

- MRS Manager: To set the alarm threshold, alarm severity, and alarm triggering time segment, choose **System > Configure Alarm Threshold** on MRS Manager.FusionInsight Manager: Choose **O&M** > **Alarm** > **Thresholds** to set the alarm threshold, alarm severity, and alarm triggering time range.
- Metrics related to Hive running can be viewed on the Hive monitoring interface.

16.10.19 "authentication failed" Is Displayed During an Attempt to Connect to the Shell Client

Symptom

In clusters in security mode, the **beeline** command fails to be executed on the Shell client when the HiveServer service is normal, and the system prompts "authentication failed". The following information is displayed.

```
Debug is true storeKey false useTicketCache true useKeyTab false doNotPrompt false ticketCache is null isInitiator true KeyTab is null refreshKrb5Config is false principal is null tryFirstPass is false useFirstPass is false storePass is false clearPass is false
Acquire TGT from Cache
Credentials are no longer valid
Principal is null
null credentials from Ticket Cache
[Krb5LoginModule] authentication failed
No password provided
```

Cause Analysis

- The client user does not perform security authentication.
- Kerberos authentication expired.

Solution

Step 1 Log in to the node where the Hive client is installed.

Step 2 Run the **source** *Cluster client installation directory*/**bigdata_env** command.

Run the **klist** command to check whether there is a valid ticket in the local end. The following information shows that the ticket became valid at 14:11:42 on December 24, 2016, and expired at 14:11:40 on December 25, 2016. In the period of time, the ticket was available.

```
klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: xxx@HADOOP.COM
Valid starting     Expires            Service principal
12/24/16 14:11:42  12/25/16 14:11:40  krbtgt/HADOOP.COM@HADOOP.COM
```

Step 3 Run the **kinit** ***username*** command for authentication and log in to the client again.

----End

16.10.20 Failed to Access ZooKeeper from the Client

Symptom

In clusters in security mode, when the HiveServer service is normal and SQL is executed by using the JDBC interface to connect to HiveServer, "The ZooKeeper client is AuthFailed" is reported.

```
14/05/19 10:52:00 WARN utils.HAClientUtilDummyWatcher: The ZooKeeper client is AuthFailed
 14/05/19 10:52:00 INFO utils.HiveHAClientUtil: Exception thrown while reading data from znode.The possible reason may be connectionless. This is recoverable. Retrying..
 14/05/19 10:52:16 WARN utils.HAClientUtilDummyWatcher: The ZooKeeper client is AuthFailed
 14/05/19 10:52:32 WARN utils.HAClientUtilDummyWatcher: The ZooKeeper client is AuthFailed
 14/05/19 10:52:32 ERROR st.BasicTestCase: Exception: Could not establish connection to active hiveserver
 java.sql.SQLException: Could not establish connection to active hiveserver
```

Or an error is reported stating "Unable to read HiveServer2 configs from ZooKeeper":

```
Exception in thread "main" java.sql.SQLException: org.apache.hive.jdbc.ZooKeeperHiveClientException: Unable to read HiveServer2 configs from ZooKeeper
at org.apache.hive.jdbc.HiveConnection.<init>(HiveConnection.java:144)
at org.apache.hive.jdbc.HiveDriver.connect(HiveDriver.java:105)
at java.sql.DriverManager.getConnection(DriverManager.java:664)
at java.sql.DriverManager.getConnection(DriverManager.java:247)
at JDBCExample.main(JDBCExample.java:82)
Caused by: org.apache.hive.jdbc.ZooKeeperHiveClientException: Unable to read HiveServer2 configs from ZooKeeper
at
org.apache.hive.jdbc.ZooKeeperHiveClientHelper.configureConnParams(ZooKeeperHiveClientHelper.java:100)
at org.apache.hive.jdbc.Utils.configureConnParams(Utils.java:509)
at org.apache.hive.jdbc.Utils.parseURL(Utils.java:429)
at org.apache.hive.jdbc.HiveConnection.<init>(HiveConnection.java:142)
... 4 more
Caused by: org.apache.zookeeper.KeeperException$ConnectionLossException: KeeperErrorCode = ConnectionLoss for /hiveserver2
at org.apache.zookeeper.KeeperException.create(KeeperException.java:99)
at org.apache.zookeeper.KeeperException.create(KeeperException.java:51)
at org.apache.zookeeper.ZooKeeper.getChildren(ZooKeeper.java:2374)
at org.apache.curator.framework.imps.GetChildrenBuilderImpl$3.call(GetChildrenBuilderImpl.java:214)
at org.apache.curator.framework.imps.GetChildrenBuilderImpl$3.call(GetChildrenBuilderImpl.java:203)
at org.apache.curator.RetryLo, op.callWithRetry(RetryLoop.java:107)
at
org.apache.curator.framework.imps.GetChildrenBuilderImpl.pathInForeground(GetChildrenBuilderImpl.java:200)
at org.apache.curator.framework.imps.GetChildrenBuilderImpl.forPath(GetChildrenBuilderImpl.java:191)
at org.apache.curator.framework.imps.GetChildrenBuilderImpl.forPath(GetChildrenBuilderImpl.java:38)
```

Cause Analysis

- When the client connects to HiveServer, the HiveServer address is automatically obtained from ZooKeeper. If ZooKeeper connection authentication is abnormal, the HiveServer address cannot be obtained from ZooKeeper correctly.
- During ZooKeeper connection authentication, **krb5.conf**, **principal**, **keytab**, and related information must be loaded to the client. Authentication failure causes are as follows:
  - The **user.keytab** path is incorrectly entered.
  - **user.principal** is incorrectly entered.
  - The cluster has switched the domain name. However, the old principal is used when the client combines the URL.

- The client cannot pass Kerberos authentication due to firewall settings. Ports 21730 (TCP), 21731 (TCP/UDP), and 21732 (TCP/UDP) need to be opened for Kerberos.

Solution

Step 1 Ensure that the user can properly access the **user.keytab** file in related paths on the client node.

Step 2 Ensure that the user's **user.principal** corresponds to the specified **keytab** file.

Run the **klist -kt keytabpath/user.keytab** command to check the file.

Step 3 If the cluster has switched the domain name, the **principal** field used in the URL must be the new domain name.

For example, the default value is **hive/hadoop.hadoop.com@HADOOP.COM**. If the cluster has switched the domain name, the field must be changed accordingly. For example, if the domain name is **abc.com**, enter **hive/hadoop.abc.com@ABC.COM**.

Step 4 Ensure that authentication is normal and HiveServer can be connected.

Run the following commands on the client:

source *Client installation directory*/**bigdata_env**

kinit *username*

Run the **beeline** command on the client to ensure normal running.

----End

16.10.21 "Invalid function" Is Displayed When a UDF Is Used

Symptom

When a UDF is created on the Hive client using Spark, "Error 10011" indicating "invalid function" is reported:

```
Error: Error while compiling statement: FAILED: SemanticException [Error 10011]: Line 1:7 Invalid function 'test_udf' (state=42000,code=10011)
```

The preceding problem occurs when multiple HiveServers use a UDF. For example, if metadata is not synchronized in time when the UDF created on HiveServer2 is used on HiveServer1, the preceding error is reported when clients on HiveServer1 are connected.

Cause Analysis

Metadata shared by multiple HiveServers or Hive and Spark is not synchronized, causing memory data inconsistency between different HiveServer instances and invalid UDF.

Solution

Synchronize new UDF information to HiveServer and reload the function.

16.10.22 Hive Service Status Is Unknown

Cause Analysis

The Hive service stops.

Solution

Restart the Hive service.

16.10.23 Health Status of a HiveServer or MetaStore Instance Is Unknown

Symptom

The health status of a HiveServer or MetaStore instance is unknown.

Cause Analysis

The HiveServer or MetaStore instance is stopped.

Solution

Restart the HiveServer or MetaStore instance.

16.10.24 Health Status of a HiveServer or MetaStore Instance Is Concerning

Symptom

The health status of the HiveServer or MetaStore instance is **Concerning**.

Cause Analysis

The HiveServer or MetaStore instance cannot be normally started. For example, when modifying the MetaStore/HiveServer GC parameter, you can view the startup log of the corresponding process, for example, the **hiveserver.out(hadoop-omm-jar-192-168-1-18.out)** file. The following exception occurs:

Error: Could not find or load main class Xmx2048M

The preceding information indicates that **Xmx2048M** is used as the startup parameter of the Java process instead of the JVM during the startup of the Java virtual machine. As shown in the following information, the hyphen (-) is deleted mistakenly.

```
METASTORE_GC_OPTS=Xms1024M Xmx2048M -DIgnoreReplayReqDetect
-XX\:CMSFullGCsBeforeCompaction\=1 -XX\:+UseConcMarkSweepGC
-XX\:+CMSParallelRemarkEnabled -XX\:+UseCMSCompactAtFullCollection
-XX\:+ExplicitGCInvokesConcurrent -server -XX\:MetaspaceSize\=128M
-XX\:MaxMetaspaceSize\=256M
```

Solution

Check the latest changes to detect incorrect settings.

```
METASTORE_GC_OPTS=Xms1024M -Xmx2048M -DIgnoreReplayReqDetect
-XX\:CMSFullGCsBeforeCompaction\=1 -XX\:+UseConcMarkSweepGC
-XX\:+CMSParallelRemarkEnabled -XX\:+UseCMSCompactAtFullCollection
-XX\:+ExplicitGCInvokesConcurrent -server -XX\:MetaspaceSize\=128M
-XX\:MaxMetaspaceSize\=256M
```

16.10.25 Garbled Characters Returned upon a select Query If Text Files Are Compressed Using ARC4

Symptom

If a Hive query result table is compressed and stored using the ARC4 algorithm, garbled characters are returned after the select * query is conducted in the result table.

Cause Analysis

The default Hive compression format is not ARC4 or output compression is disabled.

Solution

Step 1 If garbled characters are returned after the SETECT query, set the following in Beeline:

set mapreduce.output.fileoutputformat.compress.codec=org.apache.hadoop.io.encryption.arc4.ARC4BlockCodec;

set hive.exec.compress.output=true;

Step 2 Import the table to a new table using block decompression.

insert overwrite table tbl_result select * from tbl_source;

Step 3 Perform the query again.

select * from tbl_result;

----End

16.10.26 Hive Task Failed to Run on the Client But Successful on Yarn

Symptom

When Hive task running fails, an error similar to the following is reported on the client:

```
Error:Invalid OperationHandler:OperationHander [opType=EXECUTE_STATEMENT,getHandleIdentifier()=XXX]
(state=,code=0)
```

However, the MapReduce task that is submitted by the task to Yarn is successfully executed.

```
0: jdbc:hive2://189.120.204.104:21066/> select count(*) from test1;
INFO  : Number of reduce tasks determined at compile time: 1
INFO  : In order to change the average load for a reducer (in bytes):
INFO  :   set hive.exec.reducers.bytes.per.reducer=<number>
INFO  : In order to limit the maximum number of reducers:
INFO  :   set hive.exec.reducers.max=<number>
INFO  : In order to set a constant number of reducers:
INFO  :   set mapreduce.job.reduces=<number>
INFO  : number of splits:1
INFO  : Submitting tokens for job: job_1484563934624_0003
INFO  : Kind: HDFS_DELEGATION_TOKEN, Service: ha-hdfs:hacluster, Ident: (HDFS_DELEGATION_TOKEN token 7 for admin)
INFO  : Kind: HIVE_DELEGATION_TOKEN, Service: HiveServer2ImpersonationToken, Ident: 00 05 61 64 6d 69 6e 05 61 64 6d 69 6e 21 68 69 76 65 2f 68 61 64 6f 6f 70 2e 68
 85 ce e4 8a 01 59 ce 92 52 e4 8e 07 d8 0c
INFO  : The url to track the job: https://189-120-204-104:26001/proxy/application_1484563934624_0003/
INFO  : Starting Job = job_1484563934624_0003, Tracking URL = https://189-120-204-104:26001/proxy/application_1484563934624_0003/
INFO  : Kill Command = /opt/huawei/Bigdata/FusionInsight-Hive-1.1.0/hadoop/bin/hadoop job  -kill job_1484563934624_0003
INFO  : Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
INFO  : 2017-01-17 11:46:12,579 Stage-1 map = 0%,  reduce = 0%
INFO  : 2017-01-17 11:46:23,243 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.32 sec
Error: Invalid OperationHandle: OperationHandle [opType=EXECUTE_STATEMENT, getHandleIdentifier()=386323de-dfla-4299-826e-96368d4baf80] (state=,code=0)
0: jdbc:hive2://189.120.204.215:21066/>
```

Cause Analysis

The cluster where the error occurs has two HiveServer instances. The error in the log of one HiveServer instance is the same as the error (Error: Invalid OperationHandler) reported on the client. In the log of the other HiveServer instance, **START_UP** information similar to the following is printed when the error occurs, which indicates that the process is killed and restarted during that time. Because the HiveServer instance the task process plans to connect to is killed, it connects to the other healthy one, causing the error.

```
2017-02-15 14:40:11,309 | INFO  | main | STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting HiveServer2
STARTUP_MSG:   host = XXX-120-85-154/XXX.120.85.154
STARTUP_MSG:   args = []
STARTUP_MSG:   version = 1.3.0
```

Solution

Submit the task again and ensure that the HiveServer process is not manually restarted during task execution.

16.10.27 An Error Is Reported When the select Statement Is Executed

Symptom

When the **select count(*) from XXX** statement is executed, the client reports the error "Error:Error while processing statement :FAILED:Execution Error,return code 2 from...".

return code 2 indicates that the task fails because an error is reported during the execution of the MapReduce task.

```
0: jdbc:hive2://134.160.37.21:21066/> select count(*) from src.gn_data_info_gz where day_id='18' and timenap='10';
INFO  : Number of reduce tasks determined at compile time: 1
INFO  : In order to change the average load for a reducer (in bytes):
INFO  :   set hive.exec.reducers.bytes.per.reducer=<number>
INFO  : In order to limit the maximum number of reducers:
INFO  :   set hive.exec.reducers.max=<number>
INFO  : In order to set a constant number of reducers:
INFO  :   set mapreduce.job.reduces=<number>
INFO  : number of splits:496
INFO  : Submitting tokens for job: job_1482323187492_57815
INFO  : Kind: HDFS_DELEGATION_TOKEN, Service: ha-hdfs:hacluster, Ident: (HDFS_DELEGATION_TOKEN token 1083948 for boncusermm)
INFO  : Kind: HIVE_DELEGATION_TOKEN, Service: HiveServer2ImpersonationToken, Ident: 00 0a 62 6f 6e 63 75 73 65 72 6d 6d 0a 62 6f 6e 63 75 73 65 72 6d 6d 21 68 69
74 55 8a 01 59 d4 b5 f8 55 8d 02 50 ae 8e 03 65
INFO  : The url to track the job: https://hnocno3:26001/proxy/application_1482323187492_57815/
INFO  : Starting Job = job_1482323187492_57815, Tracking URL = https://hnocno3:26001/proxy/application_1482323187492_57815/
INFO  : Kill Command = /opt/huawei/Bigdata/FusionInsight_V100R002C60U10/FusionInsight-Hive-1.3.0/hive-1.3.0/bin/../../hadoop/bin/hadoop job  -kill job_148232318
INFO  : Hadoop job information for Stage-1: number of mappers: 496; number of reducers: 1
INFO  : 2017-01-18 16:21:00,906 Stage-1 map = 0%,  reduce = 0%
INFO  : 2017-01-18 16:21:18,957 Stage-1 map = 1%,  reduce = 0%, Cumulative CPU 50.53 sec
INFO  : 2017-01-18 16:21:32,526 Stage-1 map = 2%,  reduce = 0%, Cumulative CPU 416.23 sec
INFO  : 2017-01-18 16:21:35,035 Stage-1 map = 5%,  reduce = 0%, Cumulative CPU 1421.09 sec
INFO  : 2017-01-18 16:21:36,331 Stage-1 map = 7%,  reduce = 0%, Cumulative CPU 2159.35 sec
INFO  : 2017-01-18 16:21:37,810 Stage-1 map = 9%,  reduce = 0%, Cumulative CPU 2548.77 sec
INFO  : 2017-01-18 16:21:39,126 Stage-1 map = 15%,  reduce = 0%, Cumulative CPU 3264.95 sec
INFO  : 2017-01-18 16:21:40,509 Stage-1 map = 20%,  reduce = 0%, Cumulative CPU 3621.79 sec
INFO  : 2017-01-18 16:21:41,710 Stage-1 map = 26%,  reduce = 0%, Cumulative CPU 3913.79 sec
INFO  : 2017-01-18 16:21:42,890 Stage-1 map = 32%,  reduce = 0%, Cumulative CPU 4202.18 sec
INFO  : 2017-01-18 16:21:44,037 Stage-1 map = 41%,  reduce = 0%, Cumulative CPU 4595.63 sec
INFO  : 2017-01-18 16:21:45,119 Stage-1 map = 49%,  reduce = 0%, Cumulative CPU 4822.15 sec
INFO  : 2017-01-18 16:21:46,213 Stage-1 map = 57%,  reduce = 0%, Cumulative CPU 5107.44 sec
INFO  : 2017-01-18 16:21:47,300 Stage-1 map = 68%,  reduce = 0%, Cumulative CPU 5405.71 sec
INFO  : 2017-01-18 16:21:48,407 Stage-1 map = 76%,  reduce = 0%, Cumulative CPU 5611.73 sec
INFO  : 2017-01-18 16:21:49,483 Stage-1 map = 85%,  reduce = 0%, Cumulative CPU 5804.64 sec
INFO  : 2017-01-18 16:21:50,565 Stage-1 map = 92%,  reduce = 0%, Cumulative CPU 5958.81 sec
INFO  : 2017-01-18 16:21:51,641 Stage-1 map = 96%,  reduce = 0%, Cumulative CPU 6041.06 sec
INFO  : 2017-01-18 16:21:52,744 Stage-1 map = 98%,  reduce = 0%, Cumulative CPU 6073.82 sec
INFO  : 2017-01-18 16:22:08,352 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 6078.4 sec
INFO  : MapReduce Total cumulative CPU time: 0 days 1 hours 41 minutes 18 seconds 400 msec
ERROR : Ended Job = job_1482323187492_57815 with errors
Error: Error while processing statement: FAILED: Execution Error, return code 2 from org.apache.hadoop.hive.ql.exec.mr.MapRedTask (state=08S01,code=2)
0: jdbc:hive2://134.160.37.21:21066/>
```

Cause Analysis

1. Go to the native Yarn page to check the MapReduce task logs. The check result shows that the error occurs due to unidentified compression mode. The file name suffix is **.gzip** but the stack reports **.zlib**.

```
2017-01-18 16:22:07,566 INFO [main] org.apache.hadoop.hive.ql.exec.MapOperator: 4 Close done
2017-01-18 16:22:07,572 WARN [main] org.apache.hadoop.mapred.YarnChild: Exception running child : java.io.IOException: java.io.IOException: unknown compression method
        at org.apache.hadoop.hive.io.HiveIOExceptionHandlerChain.handleRecordReaderNextException(HiveIOExceptionHandlerChain.java:121)
        at org.apache.hadoop.hive.io.HiveIOExceptionHandlerUtil.handleRecordReaderNextException(HiveIOExceptionHandlerUtil.java:77)
        at org.apache.hadoop.hive.ql.io.HiveContextAwareRecordReader.doNext(HiveContextAwareRecordReader.java:355)
        at org.apache.hadoop.hive.ql.io.HiveRecordReader.doNext(HiveRecordReader.java:79)
        at org.apache.hadoop.hive.ql.io.HiveRecordReader.doNext(HiveRecordReader.java:33)
        at org.apache.hadoop.hive.ql.io.HiveContextAwareRecordReader.next(HiveContextAwareRecordReader.java:116)
        at org.apache.hadoop.mapred.MapTask$TrackedRecordReader.moveToNext(MapTask.java:199)
        at org.apache.hadoop.mapred.MapTask$TrackedRecordReader.next(MapTask.java:185)
        at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:52)
        at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:453)
        at org.apache.hadoop.mapred.MapTask.run(MapTask.java:343)
        at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:180)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1726)
        at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:174)
Caused by: java.io.IOException: unknown compression method
        at org.apache.hadoop.io.compress.zlib.ZlibDecompressor.inflateBytesDirect(Native Method)
        at org.apache.hadoop.io.compress.zlib.ZlibDecompressor.decompress(ZlibDecompressor.java:225)
        at org.apache.hadoop.io.compress.DecompressorStream.decompress(DecompressorStream.java:91)
        at org.apache.hadoop.io.compress.DecompressorStream.read(DecompressorStream.java:85)
        at java.io.InputStream.read(InputStream.java:101)
        at org.apache.hadoop.util.LineReader.fillBuffer(LineReader.java:180)
        at org.apache.hadoop.util.LineReader.readDefaultLine(LineReader.java:216)
        at org.apache.hadoop.util.LineReader.readLine(LineReader.java:174)
        at org.apache.hadoop.mapred.LineRecordReader.next(LineRecordReader.java:248)
        at org.apache.hadoop.mapred.LineRecordReader.next(LineRecordReader.java:48)
        at org.apache.hadoop.hive.ql.io.HiveContextAwareRecordReader.doNext(HiveContextAwareRecordReader.java:350)
        ... 13 more

2017-01-18 16:22:07,576 INFO [main] org.apache.hadoop.mapred.Task: Runnning cleanup for the task
```

2. Therefore, the HDFS file corresponding to the table that is queried may be incorrect. According to the file name printed in the map log, download the file from HDFS to the local end. The file whose name is suffixed with **.gz** fails to be decompressed by running the **tar** command because its format is incorrect. Run the **file** command to check the file property. The command output shows that the file is compressed from the FAT system instead of UNIX.

```
[root@hn0dn01 ~]# ls -l *.txt.gz
-rw-r--r-- 1 root root 101966463 Jan 18 20:13 20170118095958920074010l.txt.gz
-rw-r--r-- 1 root root  90448283 Jan 18 19:55 20170118104000000074002O.txt.gz
[root@hn0dn01 ~]# file 20170118095958920074010l.txt.gz
20170118095958920074010l.txt.gz: gzip compressed data, was "20170118095958920074010l.txt", from Unix, last modified: Wed Jan 18 09:59:52 2017
[root@hn0dn01 ~]# file  20170118104000000074002O.txt.gz
20170118104000000074002O.txt.gz: gzip compressed data, from FAT filesystem (MS-DOS, OS/2, NT)
[root@hn0dn01 ~]# tar -zxvf  20170118104000000074002O.txt.gz
tar: This does not look like a tar archive
tar: Skipping to next header

gzip: stdin: decompression OK, trailing garbage ignored
tar: Child returned status 2
tar: Error is not recoverable: exiting now
[root@hn0dn01 ~]#
```

Solution

Delete the file with an incorrect format from the HDFS directory or replace it with a correct one.

16.10.28 Failed to Drop a Large Number of Partitions

Symptom

When the **drop partition** operation is performed, the following information is displayed:

```
MetaStoreClient lost connection. Attempting to reconnect. |
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.invoke(RetryingMetaStoreClient.java:187)
org.apache.thrift.transport.TTransportException
at org.apache.thrift.transport.TIOStreamTransport.read(TIOStreamTransport.java:132)
at org.apache.thrift.transport.TTransport.xxx(TTransport.java:86)
at org.apache.thrift.transport.TSaslTransport.readLength(TSaslTransport.java:376)
at org.apache.thrift.transport.TSaslTransport.readFrame(TSaslTransport.java:453)
at org.apache.thrift.transport.TSaslTransport.read(TSaslTransport.java:435)
...
```

As indicated by the MetaStore log, StackOverFlow occurs.

```
2017-04-22 01:00:58,834 | ERROR | pool-6-thread-208 | java.lang.StackOverflowError
at org.datanucleus.store.rdbms.sql.SQLText.toSQL(SQLText.java:330)
at org.datanucleus.store.rdbms.sql.SQLText.toSQL(SQLText.java:339)
at org.datanucleus.store.rdbms.sql.SQLText.toSQL(SQLText.java:339)
at org.datanucleus.store.rdbms.sql.SQLText.toSQL(SQLText.java:339)
at org.datanucleus.store.rdbms.sql.SQLText.toSQL(SQLText.java:339)
```

Cause Analysis

The processing logic of the drop partition operation is to find all the partitions that meet the conditions, combine them, and delete them together. However, because the number of partitions is too large and the data stack for deleting metadata is deep, StackOverFlow errors occur.

Solution

Delete partitions in batches.

16.10.29 Failed to Start a Local Task

Symptom

1. When operations such as JOIN are performed for a small amount of data, a local task will be started. However, the execution fails and reports the following error:

```
jdbc:hive2://10.*.*.*:21066/> select a.name ,b.sex from student a join student1 b on (a.name = b.name);
ERROR : Execution failed with exit status: 1
ERROR : Obtaining error information
ERROR :
Task failed!
Task ID:
  Stage-4
...
Error: Error while processing statement: FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.mr.MapredLocalTask (state=08S01,code=1)
...
```

2. The HiveServer log shows that the local task fails to start.

```
2018-04-25 16:37:19,296 | ERROR | HiveServer2-Background-Pool: Thread-79 | Execution failed with exit status: 1 | org.apache.hadoop.hive.ql.session.SessionState $LogHelper.printError(SessionState.java:1016)
2018-04-25 16:37:19,296 | ERROR | HiveServer2-Background-Pool: Thread-79 | Obtaining error information | org.apache.hadoop.hive.ql.session.SessionState $LogHelper.printError(SessionState.java:1016)
2018-04-25 16:37:19,297 | ERROR | HiveServer2-Background-Pool: Thread-79 |
Task failed!
Task ID:
  Stage-4
Logs:
 | org.apache.hadoop.hive.ql.session.SessionState$LogHelper.printError(SessionState.java:1016)
2018-04-25 16:37:19,297 | ERROR | HiveServer2-Background-Pool: Thread-79 | /var/log/Bigdata/hive/hiveserver/hive.log | org.apache.hadoop.hive.ql.session.SessionState $LogHelper.printError(SessionState.java:1016)
2018-04-25 16:37:19,297 | ERROR | HiveServer2-Background-Pool: Thread-79 | Execution failed with exit status: 1 |
org.apache.hadoop.hive.ql.exec.mr.MapredLocalTask.executeInChildVM(MapredLocalTask.java:342)
2018-04-25 16:37:19,309 | ERROR | HiveServer2-Background-Pool: Thread-79 | FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.mr.MapredLocalTask |
org.apache.hadoop.hive.ql.session.SessionState$LogHelper.printError(SessionState.java:1016)
...
2018-04-25 16:37:36,438 | ERROR | HiveServer2-Background-Pool: Thread-88 | Error running hive query:  | org.apache.hive.service.cli.operation.SQLOperation$1$1.run(SQLOperation.java:248)
org.apache.hive.service.cli.HiveSQLException: Error while processing statement: FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.mr.MapredLocalTask
        at org.apache.hive.service.cli.operation.Operation.toSQLException(Operation.java:339)
        at org.apache.hive.service.cli.operation.SQLOperation.runQuery(SQLOperation.java:169)
        at org.apache.hive.service.cli.operation.SQLOperation.access$200(SQLOperation.java:75)
        at org.apache.hive.service.cli.operation.SQLOperation$1$1.run(SQLOperation.java:245)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1710)
        at org.apache.hive.service.cli.operation.SQLOperation$1.run(SQLOperation.java:258)
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
        at java.lang.Thread.run(Thread.java:745)
```

3. The **hs_err_pid_*****.log** file in the HiveServer log directory **/var/log/Bigdata/hive/hiveserver** contains an error about insufficient memory.

```
# There is insufficient memory for the Java Runtime Environment to continue.
# Native memory allocation (mmap) failed to map 20776943616 bytes for committing reserved memory.
  ...
```

Cause Analysis

When Hive executes JOIN for a small amount of data, MapJoin is generated. During MapJoin execution, a local task is started. JVM memory launched by the local task inherits the memory of the parent process.

When multiple JOIN operations are executed, multiple local tasks are started. If the host is out of memory, the local tasks fail to start.

Solution

Step 1 Search for the **hive.auto.convert.join** parameter and change the value of **hive.auto.convert.join** in Hive to **false**. Save the configuration and restart the service.

The value change may deteriorate service performance. You can perform the next step to avoid adverse impacts on the performance.

Step 2 Search for the **HIVE_GC_OPTS** parameter and decrease the value of **Xms** based on service requirements. The minimum value is half that of **Xmx**. After the modification, save the configuration and restart the service.

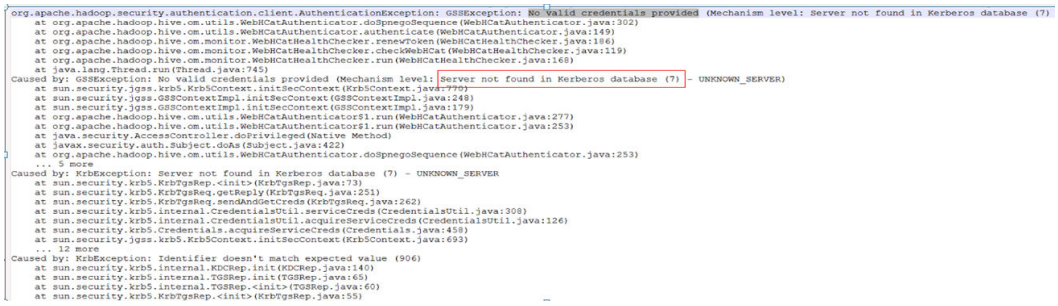
----End

16.10.30 Failed to Start WebHCat

Symptom

WebHCat fails to be started after the hostname is changed.

The following error is reported in the WebHCat startup log (**/var/log/Bigdata/hive/webhcat/hive.log**) of the corresponding node:



Cause Analysis

1. The server account of the MRS WebHCat role involves the hostname. If you change the hostname after the installation, WebHCat fails to start.
2. The one-to-many or many-to-one association between IP addresses and hostnames is configured in the **/etc/hosts** file. As a result, the IP address and hostname cannot be obtained correctly after the **hostname** and **hostname -i** commands are executed.

Solution

Step 1 Change the hostname of the modified node to the hostname before the cluster is installed.

Step 2 Check whether the **/etc/hosts** of the node where WebHCat is located is correctly configured.

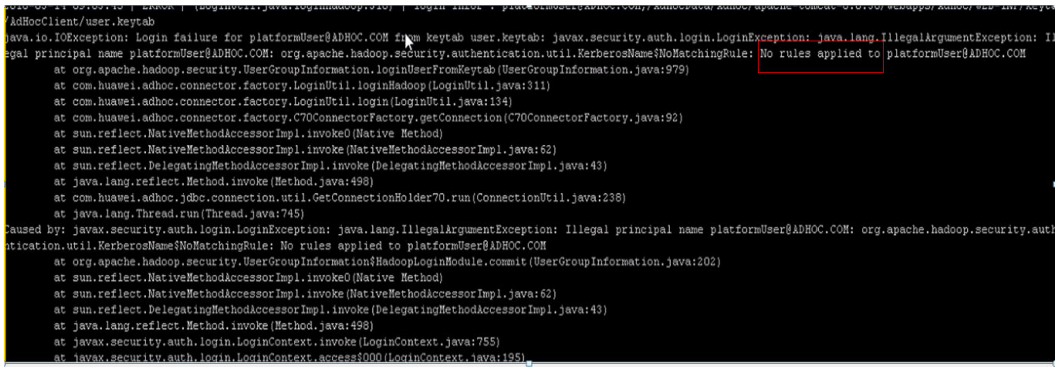
Step 3 Restart WebHCat.

----End

16.10.31 Sample Code Error for Hive Secondary Development After Domain Switching

Symptom

In the sample code for Hive secondary development, an error "No rules applied to ****" is reported:



Cause Analysis

1. The sample code for Hive secondary development loads **core-site.xml** file that is loaded through classload by default. Therefore, you need to put the configuration file to the **classpath** directory of the startup program.
2. If the domain name of the cluster is changed, the **core-site.xml** file will change. You need to download the latest **core-site.xml** file and save it to the **classpath** directory where the sample code for Hive secondary development is located.

Solution

Step 1 Download the latest client of the Hive cluster to obtain the latest **core-site.xml** file.

Step 2 Save the **core-site.xml** file to the **classpath** directory where the sample code process for Hive secondary development is located.

----End

16.10.32 MetaStore Exception Occurs When the Number of DBService Connections Exceeds the Upper Limit

Symptom

By default, the maximum number of connections to DBService is 300. If the number of connections is greater than 300 due to heavy traffic, an exception occurs in MetaStore and error "slots are reserved for non-replication superuser connections" is reported.

```
2018-04-26 14:58:55,657 | ERROR | BoneCP-pool-watch-thread | Failed to acquire connection to jdbc:postgresql://10.*.*.*:20051/hivemeta?socketTimeout=60. Sleeping for 1000 ms. Attempts left: 9 | com.jolbox.bonecp.BoneCP.obtainInternalConnection(BoneCP.java:292)
org.postgresql.util.PSQLException: FATAL: remaining connection slots are reserved for non-replication superuser connections
    at org.postgresql.core.v3.ConnectionFactoryImpl.readStartupMessages(ConnectionFactoryImpl.java:643)
    at org.postgresql.core.v3.ConnectionFactoryImpl.openConnectionImpl(ConnectionFactoryImpl.java:184)
    at org.postgresql.core.ConnectionFactory.openConnection(ConnectionFactory.java:64)
    at org.postgresql.jdbc2.AbstractJdbc2Connection.<init>(AbstractJdbc2Connection.java:124)
    at org.postgresql.jdbc3.AbstractJdbc3Connection.<init>(AbstractJdbc3Connection.java:28)
    at org.postgresql.jdbc3g.AbstractJdbc3gConnection.<init>(AbstractJdbc3gConnection.java:20)
    at org.postgresql.jdbc4.AbstractJdbc4Connection.<init>(AbstractJdbc4Connection.java:30)
    at org.postgresql.jdbc4.Jdbc4Connection.<init>(Jdbc4Connection.java:22)
    at org.postgresql.Driver.makeConnection(Driver.java:392)
    at org.postgresql.Driver.connect(Driver.java:266)
```

```
at java.sql.DriverManager.getConnection(DriverManager.java:664)
at java.sql.DriverManager.getConnection(DriverManager.java:208)
at com.jolbox.bonecp.BoneCP.obtainRawInternalConnection(BoneCP.java:361)
at com.jolbox.bonecp.BoneCP.obtainInternalConnection(BoneCP.java:269)
at com.jolbox.bonecp.ConnectionHandle.<init>(ConnectionHandle.java:242)
at com.jolbox.bonecp.PoolWatchThread.fillConnections(PoolWatchThread.java:115)
at com.jolbox.bonecp.PoolWatchThread.run(PoolWatchThread.java:82)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
at java.lang.Thread.run(Thread.java:745)
```

Cause Analysis

Heavy service traffic causes more than 300 connections to DBService, and the maximum number of connections to DBService needs to be increased.

Solution

Step 1 Search for **dbservice.database.max.connections** and set it to a proper value not greater than **1000**.

Step 2 Save the configuration and restart the affected services or instances.

Step 3 If the fault persists, check the service code for any connection leaks.

----End

16.10.33 "Failed to execute session hooks: over max connections" Reported by Beeline

Symptom

The default maximum connections to HiveServer are 200. When the number of connections exceeds 200, Beeline reports error "Failed to execute session hooks: over max connections."

```
beeline> [root@172-27-16-38 c70client]# beeline
Connecting to
jdbc:hive2://129.188.82.38:24002,129.188.82.36:24002,129.188.82.35:24002/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2;sasl.qop=auth-conf;auth=KERBEROS;principal=hive/hadoop.hadoop.com@HADOOP.COM
Debug is  true storeKey false useTicketCache true useKeyTab false doNotPrompt false ticketCache is null isInitiator true KeyTab is null refreshKrb5Config is false principal is null tryFirstPass is false useFirstPass is false storePass is false clearPass is false
Acquire TGT from Cache
Principal is xxx@HADOOP.COM
Commit Succeeded

Error: Failed to execute session hooks: over max connections. (state=,code=0)
Beeline version 1.2.1 by Apache Hive
```

The HiveServer log (**/var/log/Bigdata/hive/hiveserver/hive.log**) shows that error "over max connections" is reported.

```
2018-05-03 04:31:56,728 | WARN  | HiveServer2-Handler-Pool: Thread-137 | Error opening session:  |
org.apache.hive.service.cli.thrift.ThriftCLIService.OpenSession(ThriftCLIService.java:542)
org.apache.hive.service.cli.HiveSQLException: Failed to execute session hooks: over max connections.
    at org.apache.hive.service.cli.session.SessionManager.openSession(SessionManager.java:322)
    at org.apache.hive.service.cli.CLIService.openSessionWithImpersonation(CLIService.java:189)
    at org.apache.hive.service.cli.thrift.ThriftCLIService.getSessionHandle(ThriftCLIService.java:663)
    at org.apache.hive.service.cli.thrift.ThriftCLIService.OpenSession(ThriftCLIService.java:527)
```

```
    at org.apache.hive.service.cli.thrift.TCLIService$Processor$OpenSession.getResult(TCLIService.java:1257)
    at org.apache.hive.service.cli.thrift.TCLIService$Processor$OpenSession.getResult(TCLIService.java:1242)
    at org.apache.thrift.ProcessFunction.process(ProcessFunction.java:39)
    at org.apache.thrift.TBaseProcessor.process(TBaseProcessor.java:39)
    at org.apache.hadoop.hive.thrift.HadoopThriftAuthBridge$Server
$TUGIAssumingProcessor.process(HadoopThriftAuthBridge.java:710)
    at org.apache.thrift.server.TThreadPoolServer$WorkerProcess.run(TThreadPoolServer.java:286)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
    at java.lang.Thread.run(Thread.java:745)
Caused by: org.apache.hive.service.cli.HiveSQLException: over max connections.
    at
org.apache.hadoop.hive.transporthook.SessionControllerTsaslTransportHook.checkTotalSessionNumber(Sessi
onControllerTsaslTransportHook.java:208)
    at
org.apache.hadoop.hive.transporthook.SessionControllerTsaslTransportHook.postOpen(SessionControllerTsasl
TransportHook.java:163)
    at
org.apache.hadoop.hive.transporthook.SessionControllerTsaslTransportHook.run(SessionControllerTsaslTransp
ortHook.java:134)
    at org.apache.hive.service.cli.session.SessionManager.executeSessionHooks(SessionManager.java:432)
    at org.apache.hive.service.cli.session.SessionManager.openSession(SessionManager.java:314)
    ... 12 more
```

Cause Analysis

Heavy service traffic causes the number of connections to one HiveServer node to exceed 200, and the maximum number of connections to HiveServer needs to be increased.

Solution

Step 1 Search for **hive.server.session.control.maxconnections** and set it to a proper value not greater than **1000**.

Step 2 Save the configuration and restart the affected services or instances.

----End

16.10.34 beeline Reports the "OutOfMemoryError" Error

Symptom

When a large amount of data is queried on the Beeline client, the message "OutOFMemoryError: Java heap space" is displayed. The detailed error information is as follows:

```
org.apache.thrift.TException: Error in calling method FetchResults
    at org.apache.hive.jdbc.HiveConnection$SynchronizedHandler.invoke(HiveConnection.java:1514)
    at com.sun.proxy.$Proxy4.FetchResults(Unknown Source)
    at org.apache.hive.jdbc.HiveQueryResultSet.next(HiveQueryResultSet.java:358)
    at org.apache.hive.beeline.BufferedRows.<init>(BufferedRows.java:42)
    at org.apache.hive.beeline.BeeLine.print(BeeLine.java:1856)
    at org.apache.hive.beeline.Commands.execute(Commands.java:873)
    at org.apache.hive.beeline.Commands.sql(Commands.java:714)
    at org.apache.hive.beeline.BeeLine.dispatch(BeeLine.java:1035)
    at org.apache.hive.beeline.BeeLine.execute(BeeLine.java:821)
    at org.apache.hive.beeline.BeeLine.begin(BeeLine.java:778)
    at org.apache.hive.beeline.BeeLine.mainWithInputRedirection(BeeLine.java:486)
    at org.apache.hive.beeline.BeeLine.main(BeeLine.java:469)
Caused by: java.lang.OutOfMemoryError: Java heap space
    at com.sun.crypto.provider.CipherCore.doFinal(CipherCore.java:959)
    at com.sun.crypto.provider.CipherCore.doFinal(CipherCore.java:824)
```

```
    at com.sun.crypto.provider.AESCipher.engineDoFinal(AESCipher.java:436)
    at javax.crypto.Cipher.doFinal(Cipher.java:2223)
    at sun.security.krb5.internal.crypto.dk.AesDkCrypto.decryptCTS(AesDkCrypto.java:414)
    at sun.security.krb5.internal.crypto.dk.AesDkCrypto.decryptRaw(AesDkCrypto.java:291)
    at sun.security.krb5.internal.crypto.Aes256.decryptRaw(Aes256.java:86)
    at sun.security.jgss.krb5.CipherHelper.aes256Decrypt(CipherHelper.java:1397)
    at sun.security.jgss.krb5.CipherHelper.decryptData(CipherHelper.java:576)
    at sun.security.jgss.krb5.WrapToken_v2.getData(WrapToken_v2.java:130)
    at sun.security.jgss.krb5.WrapToken_v2.getData(WrapToken_v2.java:105)
    at sun.security.jgss.krb5.Krb5Context.unwrap(Krb5Context.java:1058)
    at sun.security.jgss.GSSContextImpl.unwrap(GSSContextImpl.java:403)
    at com.sun.security.sasl.gsskerb.GssKrb5Base.unwrap(GssKrb5Base.java:77)
    at org.apache.thrift.transport.TSaslTransport$SaslParticipant.unwrap(TSaslTransport.java:559)
    at org.apache.thrift.transport.TSaslTransport.readFrame(TSaslTransport.java:462)
    at org.apache.thrift.transport.TSaslTransport.read(TSaslTransport.java:435)
    at org.apache.thrift.transport.TSaslClientTransport.read(TSaslClientTransport.java:37)
    at org.apache.thrift.transport.TTransport.xxx(TTransport.java:86)
    at org.apache.hadoop.hive.thrift.TFilterTransport.xxx(TFilterTransport.java:62)
    at org.apache.thrift.protocol.TBinaryProtocol.xxx(TBinaryProtocol.java:429)
    at org.apache.thrift.protocol.TBinaryProtocol.readI32(TBinaryProtocol.java:318)
    at org.apache.thrift.protocol.TBinaryProtocol.readMessageBegin(TBinaryProtocol.java:219)
    at org.apache.thrift.TServiceClient.receiveBase(TServiceClient.java:77)
    at org.apache.hive.service.cli.thrift.TCLIService$Client.recv_FetchResults(TCLIService.java:505)
    at org.apache.hive.service.cli.thrift.TCLIService$Client.FetchResults(TCLIService.java:492)
    at sun.reflect.GeneratedMethodAccessor2.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at org.apache.hive.jdbc.HiveConnection$SynchronizedHandler.invoke(HiveConnection.java:1506)
    at com.sun.proxy.$Proxy4.FetchResults(Unknown Source)
    at org.apache.hive.jdbc.HiveQueryResultSet.next(HiveQueryResultSet.java:358)
Error: Error retrieving next row (state=,code=0)
```

Cause Analysis

- The data volume is excessively large.
- Users use the **select * from table_name;** statement for query in the whole table. There is a large amount of data in the table.
- The default startup memory of Beeline is 128 MB. The returned result set is too large during query, overloading Beeline.

Solution

Step 1 Before running **select count(*) from table_name;**, check the amount of data to be queried and determine whether to display data of this magnitude in Beeline.

Step 2 If a certain amount of data needs to be displayed, adjust the JVM parameter of the Hive client. Add **export HIVE_OPTS=-Xmx1024M** (change the value based on service requirements) to **component_env** in the **/Hive** directory of the Hive client. Run the **source** command to obtain the **/bigdata_env** directory on the client.

----End

16.10.35 Task Execution Fails Because the Input File Number Exceeds the Threshold

Symptom

When Hive performs a query operation, error message "Job Submission failed with exception 'java.lang.RuntimeException(input file number exceeded the limits in the conf;input file num is: 2380435,max heap memory is: 16892035072,the limit conf

is: 500000/4)'" is displayed. The value in the error message varies depending on the actual situation. The error details are as follows:

```
ERROR : Job Submission failed with exception 'java.lang.RuntimeException(input file numbers exceeded the limits in the conf;
 input file num is: 2380435 ,
 max heap memory is: 16892035072 ,
 the limit conf is: 500000/4)'
java.lang.RuntimeException: input file numbers exceeded the limits in the conf;
 input file num is: 2380435 ,
 max heap memory is: 16892035072 ,
 the limit conf is: 500000/4
    at org.apache.hadoop.hive.ql.exec.mr.ExecDriver.checkFileNum(ExecDriver.java:545)
    at org.apache.hadoop.hive.ql.exec.mr.ExecDriver.execute(ExecDriver.java:430)
    at org.apache.hadoop.hive.ql.exec.mr.MapRedTask.execute(MapRedTask.java:137)
    at org.apache.hadoop.hive.ql.exec.Task.executeTask(Task.java:158)
    at org.apache.hadoop.hive.ql.exec.TaskRunner.runSequential(TaskRunner.java:101)
    at org.apache.hadoop.hive.ql.Driver.launchTask(Driver.java:1965)
    at org.apache.hadoop.hive.ql.Driver.execute(Driver.java:1723)
    at org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1475)
    at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1283)
    at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1278)
    at org.apache.hive.service.cli.operation.SQLOperation.runQuery(SQLOperation.java:167)
    at org.apache.hive.service.cli.operation.SQLOperation.access$200(SQLOperation.java:75)
    at org.apache.hive.service.cli.operation.SQLOperation$1$1.run(SQLOperation.java:245)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.Subject.doAs(Subject.java:422)
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1710)
    at org.apache.hive.service.cli.operation.SQLOperation$1.run(SQLOperation.java:258)
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511)
    at java.util.concurrent.FutureTask.run(FutureTask.java:266)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
    at java.lang.Thread.run(Thread.java:745)

Error: Error while processing statement: FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.mr.MapRedTask (state=08S01,code=1)
```

Cause Analysis

MRS uses the ratio of maximum files to the maximum HiveServer heap memory to determine the number of input files allowed in a MapReduce job submission. Default value **500000/4** indicates that each 4 GB of heap memory allows a maximum of 500,000 input files. An error occurs if the number of input files exceeds this limit.

Solution

Step 1 Search for **hive.mapreduce.input.files2memory** and set it to a proper value based on the actual memory and task.

Step 2 Save the configuration and restart the affected services or instances.

Step 3 If the fault persists, adjust the GC parameter of the HiveServer based on service requirements.

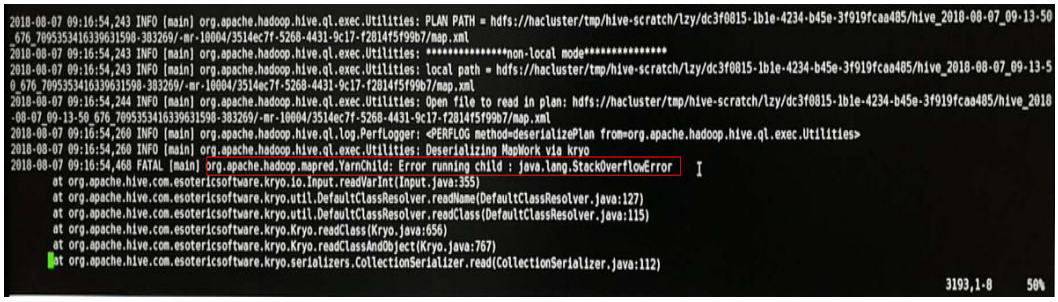
----End

16.10.36 Task Execution Fails Because of Stack Memory Overflow

Symptom

When Hive performs a query operation, error "Error running child: java.lang.StackOverflowError" is reported. The error details are as follows:

```
FATAL [main] org.apache.hadoop.mapred.YarnChild: Error running child : java.lang.StackOverflowError
at org.apache.hive.come.esotericsoftware.kryo.io.Input.readVarInt(Input.java:355)
at
org.apache.hive.come.esotericsoftware.kryo.util.DefautClassResolver.readName(DefautClassResolver.java:127)
at
org.apache.hive.come.esotericsoftware.kryo.util.DefautClassResolver.readClass(DefautClassResolver.java:115)
at org.apache.hive.come.esotericsoftware.kryo.Kryo.readClass(Kryo.java.656)
at org.apache.hive.come.esotericsoftware.kryo.kryo.readClassAnd0bject(Kryo.java:767)
at
org.apache.hive.come.esotericsoftware.kryo.serializers.collectionSerializer.read(CollectionSerializer.java:112)
```



Cause Analysis

Error "java.lang.StackOverflowError" indicates the memory overflow of the thread stack. It may occur if there are multiple levels of calls (for example, infinite recursive calls) or the thread stack is too small.

Solution

Adjust the stack memory in the JVM parameters of the Map and Reduce stages during execution of a MapReduce job, that is, **mapreduce.map.java.opts** (adjusting the stack memory of Map) and **mapreduce.reduce.java.opts** (adjusting the stack memory of Reduce). The following uses the **mapreduce.map.java.opts** parameter as an example.

- To increase the Map memory temporarily (only valid for Beeline):

  Run the **set mapreduce.map.java.opts=-Xss8G;** command on the Beeline client. (Change the value as required.)

- To permanently increase the Map memory specified by the **mapreduce.map.memory.mb** and **mapreduce.map.java.opts** parameters:

  a. Add custom parameter **mapreduce.map.java.opts** and set it to a proper value.

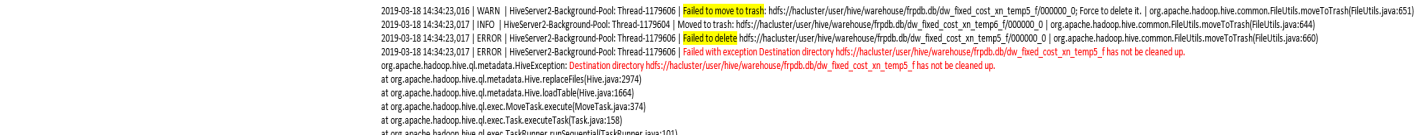
  b. Save the configuration and restart the affected services or instances.

  Note that the modification takes effect after a service restart. During the restart, the Hive service is unavailable.

16.10.37 Task Failed Due to Concurrent Writes to One Table or Partition

Symptom

When Hive executes an INSERT statement, an error is reported indicating that a file or directory already exists or is cleared in HDFS. The error details are as follows:



Cause Analysis

1. Check the start time and end time of the task based on the HiveServer audit logs.
2. Check whether data is inserted into the same table or partition in the time segment.
3. Hive does not support concurrent data insertion for a table or partition. As a result, multiple tasks perform operations on the same temporary data directory, and one task moves the data of another task, causing task failure.

Solution

The service logic is modified so that data is inserted to the same table or partition in single thread mode.

16.10.38 Hive Task Failed Due to a Lack of HDFS Directory Permission

Symptom

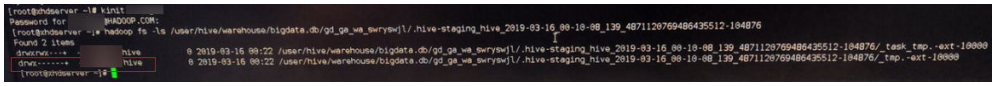
An error message is displayed, indicating that the user does not have the permission to access the HDFS directory.

```
2019-04-09 17:49:19,845 | ERROR | HiveServer2-Background-Pool: Thread-3160445 | Job Submission failed with exception 'org.apache.hadoop.security.AccessControlException(Permission denied: user=hive_quanxian, access=READ_EXECUTE, inode="/user/hive/warehouse/bigdata.db/gd_ga_wa_swryswjl":zhongao:hive:drwx------
at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkAccessAcl(FSPermissionChecker.java:426)
at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.check(FSPermissionChecker.java:329)
at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkSubAccess(FSPermissionChecker.java:300)
at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:241)
at com.xxx.hadoop.adapter.hdfs.plugin.HWAccessControlEnforce.checkPermission(HWAccessControlEnforce.java:69)
at org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionChecker.java:19
```

```
0)
at org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPermission(FSDirectory.java:1910)
at org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPermission(FSDirectory.java:1894)
at
org.apache.hadoop.hdfs.server.namenode.FSDirStatAndListingOp.getContentSummary(FSDirStatAndListingO
p.java:135)
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.getContentSummary(FSNamesystem.java:3983)
at
org.apache.hadoop.hdfs.server.namenode.NameNodeRpcServer.getContentSummary(NameNodeRpcServer.ja
va:1342)
at
org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolServerSideTranslatorPB.getContentSummary(Cli
entNamenodeProtocolServerSideTranslatorPB.java:925)
at org.apache.hadoop.hdfs.protocol.proto.ClientNamenodeProtocolProtos$ClientNamenodeProtocol
$2.callBlockingMethod(ClientNamenodeProtocolProtos.java)
at org.apache.hadoop.ipc.ProtobufRpcEngine$Server$ProtoBufRpcInvoker.call(ProtobufRpcEngine.java:616)
at org.apache.hadoop.ipc.RPC$Server.call(RPC.java:973)
at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2260)
at org.apache.hadoop.ipc.Server$Handler$1.run(Server.java:2256)
at java.security.AccessController.doPrivileged(Native Method)
at javax.security.auth.Subject.doAs(Subject.java:422)
at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1781)
at org.apache.hadoop.ipc.Server$Handler.run(Server.java:2254)
)'
```

Cause Analysis

1. According to the stack information, the permission on the subdirectory fails to be checked.
   ```
   org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkSubAccess(FSPermissionChecker.ja
   va:300)
   ```
2. Check the permission of all files and directories in HDFS. The permission of a directory is 700 (only the file owner can access the directory). It is confirmed that an abnormal directory exists.



Solution

1. Check whether the file is imported manually. If not, delete the file or directory.
2. If the file or directory cannot be deleted, change the file or directory permission to 770.

16.10.39 Failed to Load Data to Hive Tables

Symptom

After creating a table, a user runs the **LOAD** command to import data to the table. However, the following problem occurs during the import:

```
.....
> LOAD DATA INPATH '/user/tester1/hive-data/data.txt' INTO TABLE employees_info;
Error: Error while compiling statement: FAILED: SemanticException Unable to load data to destination table.
Error: The file that you are trying to load does not match the file format of the destination table.
(state=42000,code=40000)
.....
```

Cause Analysis

1. The storage format is not specified during table creation, and the default format RCFile is used.
2. However, the data to be imported is in TEXTFILE format.

Solution

This problem is caused by an application defect. You can use a proper method based on site requirements only by ensuring that the storage format specified by the table is the same as the format of the data to be imported.

- Method 1:

  Specify the storage format when creating a table as a user who has the Hive table operation permission. For example:

  CREATE TABLE IF NOT EXISTS employees_info(name STRING,age INT) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;

  Specify the format of the data to be imported as TEXTFILE.
- Method 2:

  Import RCFile data, but not TEXTFILE data.

16.10.40 HiveServer and HiveHCat Process Faults

Issue

The HiveServer and WebHCat processes in the customer cluster are faulty.

Symptom

The HiveServer and WebHCat processes on the Master2 node in the MRS cluster are faulty. After the restart, the processes are still faulty.

Cause Analysis

On Manager, start the faulty HiveServer process. Log in to the background and search for the error information at the corresponding time point in the **hiveserver.out** log file. The error information is as follows: **error parsing conf mapred-site.xml** and **Premature end of file**. Restart WebHCat. The same error is reported because the **mapred-site.xml** file fails to be parsed.

Procedure

1. Log in to the Master2 node as user **root**.
2. Run the **find / -name 'mapred-site.xml'** command to obtain the location of the **mapred-site.xml** file.
   - The path of HiveServer is **/opt/Bigdata/***Cluster version***/1_13_HiveServer/etc/mapred-site.xml**.
   - The path of WebHCat is **/opt/Bigdata/***Cluster version***/1_13_WebHCat/etc/mapred-site.xml**.

3. Check whether the **mapred-site.xml** file is normal. In this case, the configuration file is empty. As a result, the parsing fails.
4. Restore the **mapred-site.xml** file. Run the **scp** command to copy the configuration file in the corresponding directory on the Master1 node to the corresponding directory on the Master2 node to replace the original file.
5. Run the **chown omm:wheel mapred-site.xml** command to change the owner group and user.
6. On Manager, restart the faulty HiveServer and WebHCat processes.

16.10.41 An Error Occurs When the INSERT INTO Statement Is Executed on Hive But the Error Message Is Unclear

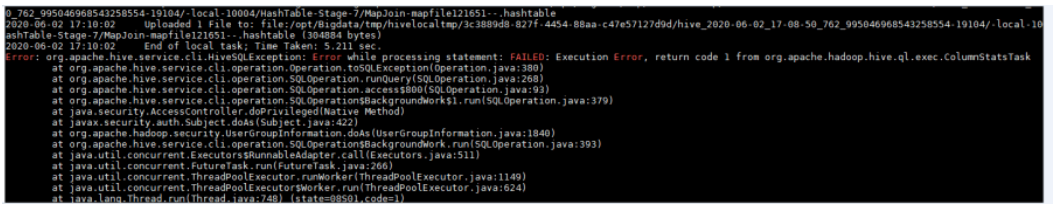
Issue

An error is reported when a user uses MRS Hive to execute a SQL statement.

Symptom

When a user uses MRS Hive to execute a SQL statement, the following error message is displayed.

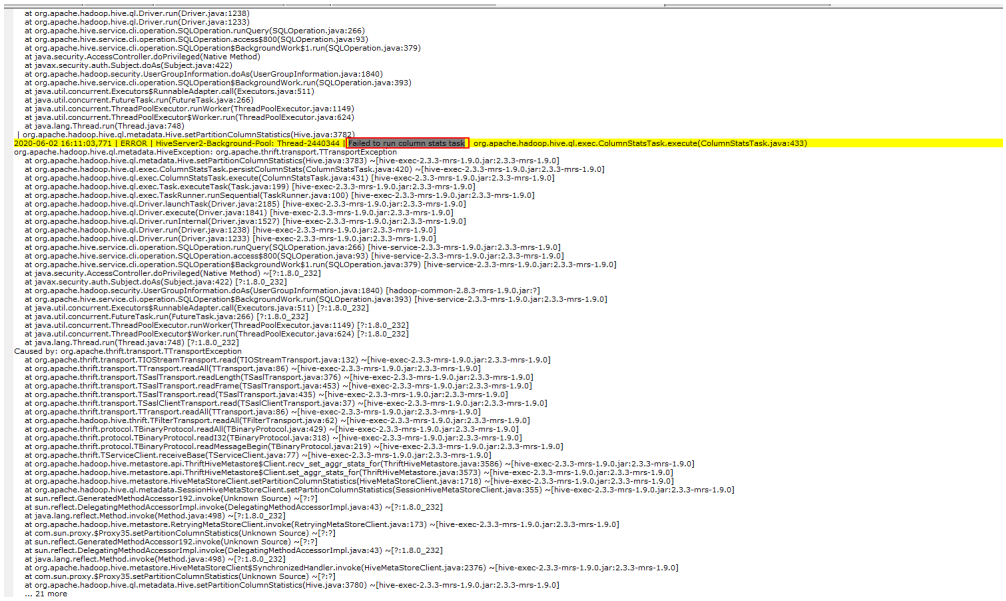
Figure 16-38 Error reported when MRS Hive executes a SQL statement



Cause Analysis

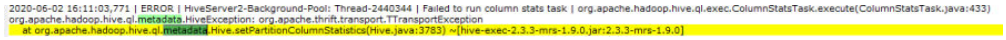
1. The HiveServer log shows the following message at the time when the error is reported.

Figure 16-39 HiveServer log



2. No important information is found in that log, but the **metadata** field is found in the stack. Therefore, the error may be related to MetaStore.

Figure 16-40 Metadata in the stack



3. The MetaStore log shows the following error information.

Figure 16-41 MetaStore log

The error context indicates that an error occurs during SQL statement execution, and the following information is displayed in the error message:

```
Caused by: org.postgresql.util.PSQLException: ERROR: value too long for type character varying(4000)
```

The SQL statement fails because the length of all columns exceeds 4000 bytes. The restriction needs to be modified.

Procedure

Step 1 Log in to any master node in the cluster as user **root** and run the **su - omm** command to switch to user **omm**.

Step 2 Run the following command to log in to GaussDB:

gsql -p 20051 -d hivemeta -U *username* -W *password*

Step 3 Run the following command to modify the restriction:

alter table PARTITION_PARAMS alter column PARAM_VALUE type varchar(6000);

----End

16.10.42 Timeout Reported When Adding the Hive Table Field

Issue

An error message is reported when adding the Hive table fields.

Symptom

Hive executes **ALTER TABLE table_name ADD COLUMNS(column_name string) CASCADE** on tables that contain more than 10,000 partitions. The error information is as follows:

```
Timeout when executing method: alter_table_with_environment_context; 600525ms exceeds 600000ms
```

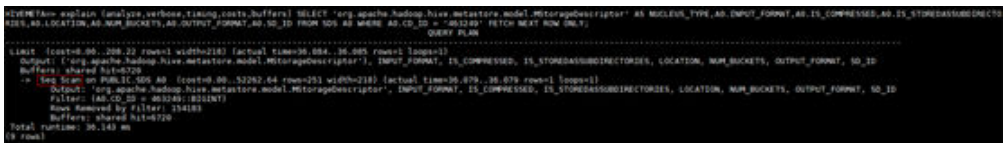
Cause Analysis

1. The MetaStore client connection times out. The default timeout interval for the connection between the MetaStore client and server is 600 seconds. On FusionInsight Manager, increase the value of **hive.metastore.client.socket.timeout** to **3600s**.
2. Another error is reported:
   ```
   Error: org.apache.hive.service.cli.HiveSQLException: Error while processing statement: FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask. Unable to alter table. java.net.SocketTimeoutException: Read timed out
   ```
   JDBC connection timeout interval of the MetaStore metadata. The default value is 60 ms.
3. Increase the value of **socketTimeout** in **javax.jdo.option.ConnectionURL** to **60000**. The initial error is still reported.
   ```
   Timeout when executing method: alter_table_with_environment_context;3600556ms exceeds 3600000ms
   ```
4. Increase the values of parameters such as **hive.metastore.batch.retrieve.max**, **hive.metastore.batch.retrieve.table.partition.max**, and **dbservice.database.max.connections**. The problem persists.
5. It is suspected that the problem is caused by the GaussDB because adding a field will traverse each partition to execute **getPartitionColumnStatistics** and **alterPartition**.
6. Run the **gsql -p 20051 -U omm -W dbserverAdmin@123 -d hivemeta** command as user **omm** to log in to the Hive metabase.

7. Run **select * from pg_locks**. No lock wait is found.
8. Run **select * from pg_stat_activity;**. It is found that the process execution takes a long time.

```
SELECT 'org.apache.hadoop.hive.metastore.model.MPartitionColumnStatistics'AS
NUCLEUS_TYPE,A0.AVG_COL_LEN,A0."COLUMN_NAME",A0.COLUMN_TYPE,A0.DB_NAME,A0.BIG_DECI
MAL_HIGH_VALUE,A0.BIG_DECIMAL_LOW_VALUE,A0.DOUBLE_HIGH_VALUE,A0.DOUBLE_LOW_VALUE,
A0.LAST_ANALYZED,A0.LONG_HIGH_VALUE,A0.LONG_LOW_VALUE,A0.MAX_COL_LEN,A0.NUM_DISTIN
CTS,A0.NUM_FALSES,A0.NUM_NULLS,A0.NUM_TRUES,A0.PARTITION_NAME,A0."TABLE_NAME",A0.CS_
ID,A0.PARTITION_NAMEAS NUCORDER0 FROM PART_COL_STATS A0 WHERE A0."TABLE_NAME" =
'$1' ANDA0.DB_NAME = '$2' AND A0.PARTITION_NAME = '$3' AND((((((A0."COLUMN_NAME" = '$4')
OR (A0."COLUMN_NAME" ='$5')) OR (A0."COLUMN_NAME" = '$6')) OR (A0."COLUMN_NAME"
='$7')) OR (A0."COLUMN_NAME" = '$8')) OR (A0."COLUMN_NAME" ='$9')) ORDER BY NUCORDER0;
```

9. Run the **gs_guc reload -c log_min_duration_statement=100 -D /srv/BigData/dbdata_service/data/** command to start SQL recording. It is found that the execution duration of the **Run select * from pg_sta...** statement is **700 ms**, and more than 10,000 commands are executed because there are more than 10,000 partitions.
10. Add explain (analyze, verbose, timing, costs, buffers) before the SQL statement to analyze the execution plan. It is found that the entire table needs to be scanned during execution.



11. Check the index. It is found that the index does not meet the leftmost match rule.

```
HIVEMETA=# \d+ PART_COL_STATS
                                    Table "PUBLIC.PART_COL_STATS"
        Column         |          Type          |          Modifiers          | Storage  | Stats target | Description
-----------------------+------------------------+-----------------------------+----------+--------------+-------------
 CS_ID                 | BIGINT                 | not null                    | plain    |              |
 CAT_NAME              | CHARACTER VARYING(256) | default NULL::CHARACTER VARYING | extended |              |
 DB_NAME               | CHARACTER VARYING(128) | default NULL::CHARACTER VARYING | extended |              |
 TABLE_NAME            | CHARACTER VARYING(256) | default NULL::CHARACTER VARYING | extended |              |
 PARTITION_NAME        | CHARACTER VARYING(767) | default NULL::CHARACTER VARYING | extended |              |
 COLUMN_NAME           | CHARACTER VARYING(767) | default NULL::CHARACTER VARYING | extended |              |
 COLUMN_TYPE           | CHARACTER VARYING(128) | default NULL::CHARACTER VARYING | extended |              |
 PART_ID               | BIGINT                 | not null                    | plain    |              |
 LONG_LOW_VALUE        | BIGINT                 |                             | plain    |              |
 LONG_HIGH_VALUE       | BIGINT                 |                             | plain    |              |
 DOUBLE_LOW_VALUE      | DOUBLE PRECISION       |                             | plain    |              |
 DOUBLE_HIGH_VALUE     | DOUBLE PRECISION       |                             | plain    |              |
 BIG_DECIMAL_LOW_VALUE | CHARACTER VARYING(4000) | default NULL::CHARACTER VARYING | extended |              |
 BIG_DECIMAL_HIGH_VALUE | CHARACTER VARYING(4000) | default NULL::CHARACTER VARYING | extended |              |
 NUM_NULLS             | BIGINT                 | not null                    | plain    |              |
 NUM_DISTINCTS         | BIGINT                 |                             | plain    |              |
 BIT_VECTOR            | BYTEA                  |                             | extended |              |
 AVG_COL_LEN           | DOUBLE PRECISION       |                             | plain    |              |
 MAX_COL_LEN           | BIGINT                 |                             | plain    |              |
 NUM_TRUES             | BIGINT                 |                             | plain    |              |
 NUM_FALSES            | BIGINT                 |                             | plain    |              |
 LAST_ANALYZED         | BIGINT                 | not null                    | plain    |              |
Indexes:
    "PART_COL_STATS_pkey" PRIMARY KEY, BTREE (CS_ID)
    "PART_COL_STATS_N49" BTREE (PART_ID)
    "PCS_STATS_IDX" BTREE (CAT_NAME, DB_NAME, TABLE_NAME, COLUMN_NAME, PARTITION_NAME)
Foreign-key constraints:
    "PART_COL_STATS_fkey" FOREIGN KEY (PART_ID) REFERENCES PARTITIONS(PART_ID) DEFERRABLE
Has OIDs: no
```

Procedure

1. Rebuild an index.

   ```
   su – omm
   gsql -p 20051 -U omm -W dbserverAdmin@123 -d hivemeta
   DROP INDEX PCS_STATS_IDX;
   CREATE INDEX PCS_STATS_IDX ON PART_COL_STATS(DB_NAME, TABLE_NAME, COLUMN_NAME, PARTITION_NAME);
   CREATE INDEX SDS_N50 ON SDS(CD_ID);
   ```

2. Check the execution plan again. It is found that the statement can be indexed and executed within 5 ms (the original execution time is 700 ms). Add fields to the Hive table again. The fields can be added successfully.

16.10.43 Failed to Restart the Hive Service

Issue

After the Hive service configuration is modified, the configuration fails to be saved. The configuration status of the Hive service on Manager is **Failed**.

Symptom

User A opens the Hive configuration file in the background of the MRS node and does not close the file. User B modifies the Hive configuration item in **Service Management** on the MRS Manager page, saves the configuration, and restarts the Hive service. However, the configuration fails to be saved and the Hive service fails to be started.

Cause Analysis

When user B modifies the configuration on the MRS Manager page, the configuration file is opened by user A in the background of an MRS node. As a

result, the configuration file cannot be replaced and the Hive service fails to be started.

Procedure

Step 1 Manually close the Hive configuration file opened in the background of the cluster node.

Step 2 Modify the Hive configuration on MRS Manager and save the configuration.

Step 3 Restart the Hive service.

----End

16.10.44 Hive Failed to Delete a Table

Issue

Hive fails to delete a table.

Symptom

Partitioning a Hive table by two columns may eventually generate over 20,000 partition files. As a result, the user fails to execute the **truncate table ${TableName}** or **drop table ${TableName}** statement to delete table data.

Cause Analysis

The file deletion operations are executed by a single thread serially. If the Hive partitioned tables have too many partition files, a large amount of metadata is stored in the metadata database. It takes a long time to delete metadata when a statement is executed to delete table data. As a result, the deletion cannot be complete within the specified timeout period, and the operation fails.

NOTE

You can log in to FusionInsight Manager and choose **Cluster** > **Services** > **Hive**. On the Hive page, choose **Configuration** > **All Configurations**, choose **ServerInit** under **MetaStore(Role)** in the navigation tree, and view the **hive.metastore.client.socket.timeout** parameter value in the right pane. This value is the timeout period. You can view the default value in the **Description** column.

Procedure

Step 1 (Optional, perform this step for an internal table) Use **alter table ${TableName} set TBLPROPERTIES('EXTERNAL'='true')** to convert it into an external table. In this way, only its metadata but not data stored on the HDFS is deleted, saving the table deletion time.

Step 2 (Optional, perform this step to use the same table name) Run the **show create table ${TableName}** command to export the table structure, and then run the **ALTER TABLE ${TableName} RENAME TO ${new_table_name};** command to rename the table. In this way, you can create a table that is the same as the original one.

Step 3 Run the **hdfs dfs –rm –r –f ${hdfs_path}** command to delete table data from the HDFS.

Step 4 Use **alter table ${Table_Name} drop partition (${PartitionName}<'XXXX', ${PartitionName}>'XXXX');** in Hive to delete partitions and reduce the number of files. The deletion conditions can be flexibly configured.

Step 5 When the number of rest partitions is smaller than 1,000, run the **drop table ${TableName}** command to delete the table.

----End

Summary and Suggestions

Hive partitioning can improve query efficiency. However, you should properly plan the partitioning policies to prevent a large number of small files from being generated.

16.10.45 An Error Is Reported When msck repair table table_name Is Run on Hive

Symptom

When **msck repair table table_name** is run on Hive, the error message "FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask (state=08S01,code=1)" is displayed.

Possible Causes

A directory in the HiveServer log file **/var/log/Bigdata/hive/hiveserver/hive.log** does not comply with the partition format.

```
2020-07-15 15:38:10,427 | WARN  | HiveServer2-Background-Pool: Thread-10905216 | Failed to run metacheck: | org.apache.hadoop.hive.ql.exec.DDLTask.msck(DDLTask.java:2023)
org.apache.hadoop.hive.ql.metadata.HiveException: Repair: Cannot add partition ads_marketing.t_marketing_telemarketing_order_list:dtime=2020-04-24 17%3A55%3A00 due to invalid characters in the name
    at org.apache.hadoop.hive.ql.exec.DDLTask.msck(DDLTask.java:1986) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.exec.DDLTask.execute(DDLTask.java:424) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.exec.Task.executeTask(Task.java:199) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.exec.TaskRunner.runSequential(TaskRunner.java:100) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.Driver.launchTask(Driver.java:2185) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.Driver.execute(Driver.java:1841) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1527) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1230) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1233) [hive-exec-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hive.service.cli.operation.SQLOperation.runQuery(SQLOperation.java:266) [hive-service-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hive.service.cli.operation.SQLOperation.access$800(SQLOperation.java:93) [hive-service-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at org.apache.hive.service.cli.operation.SQLOperation$BackgroundWork$1.run(SQLOperation.java:379) [hive-service-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at java.security.AccessController.doPrivileged(Native Method) ~[?:1.8.0_232]
    at javax.security.auth.Subject.doAs(Subject.java:422) [?:1.8.0_232]
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1840) [hadoop-common-2.8.3-mrs-1.9.0.jar:?]
    at org.apache.hive.service.cli.operation.SQLOperation$BackgroundWork.run(SQLOperation.java:393) [hive-service-2.3.3-mrs-1.9.0.jar:2.3.3-mrs-1.9.0]
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) [?:1.8.0_232]
    at java.util.concurrent.FutureTask.run(FutureTask.java:266) [?:1.8.0_232]
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [?:1.8.0_232]
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [?:1.8.0_232]
    at java.lang.Thread.run(Thread.java:748) [?:1.8.0_232]
```

Procedure

- Method 1: Delete the incorrect file or directory.
- Method 2: Run the **set hive.msck.path.validation=skip** command to skip invalid directories.

16.10.46 How Do I Release Disk Space After Dropping a Table in Hive?

Issue

After a user runs the **drop** command on the Hive CLI to drop a table and then uses the **hdfs dfsadmin -report** command to check the disk space, the command output shows that the table is not deleted.

Cause Analysis

The **drop** command executed on the Hive CLI deletes only the table structure of the external table, but not the table data stored in HDFS.

Procedure

Step 1 Log in to the node where the client is installed as user **root** and authenticate the component user.

cd *Client installation directory*

source bigdata_env

kinit *Component service user* (Skip this step for clusters with Kerberos authentication disabled.)

Step 2 Run the following command to delete the table stored in HDFS:

hadoop fs -rm hdfs://hacluster/*Path of the table*

----End

16.10.47 Connection Timeout During SQL Statement Execution on the Client

Symptom

The SQL statement fails to be executed on the client, and the error message "Timed out waiting for a free available connection" is displayed.

Possible Causes

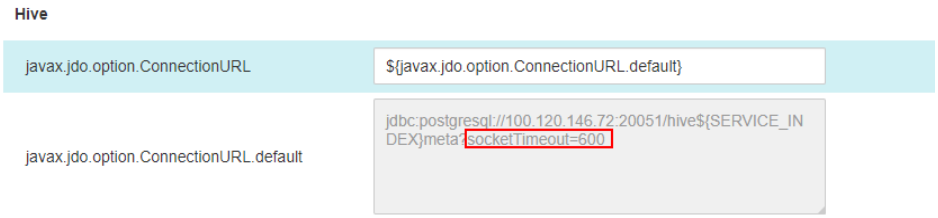
A large number of DBService connections exist, and obtaining connections times out.

Procedure

Step 1 Check whether the client uses the Spark-SQL client to execute SQL statements.

- If yes, check the timeout parameter in the URL, change the value to **600**, and go to **Step 7**.
- If the alarm is not cleared, go to **Step 2**.

Step 2 Log in to FusionInsight Manager, choose **Cluster > Services > Hive > Configuration > All Configurations**, search for **javax.jdo.option.ConnectionURL**, and check whether the value of the timeout parameter is less than **600**.



NOTE

This parameter exists in Hive, HiveServer, MetaStore, and WebHCat. Ensure that the parameter values are the same.

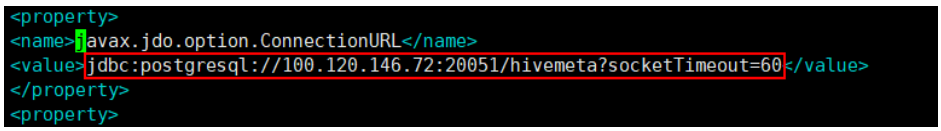
- If yes, go to **Step 3**.
- If no, go to **Step 7**.

Step 3 Check whether the value of **javax.jdo.option.ConnectionURL** is **${javax.jdo.option.ConnectionURL.default}**.

- If yes, go to **Step 4**.
- If no, change the timeout parameter in the URL to **600**, click **Save**, and go to **Step 7**.

Step 4 Click **Instance**, select any HiveServer instance, and log in to the instance node as user **root**.

Step 5 Open the **${BIGDATA_HOME}/FusionInsight_Current/*HiveServer/etc/hivemetastore-site.xml** configuration file, find the **javax.jdo.option.ConnectionURL** parameter, and copy its value.



Step 6 Log in to FusionInsight Manager, choose **Cluster > Services > Hive > Configuration > All Configurations**, search for **javax.jdo.option.ConnectionURL**, change its value to the URL copied in **Step 5**, change the timeout parameter to **600**, and click **Save**.

NOTE

This parameter exists in Hive, HiveServer, MetaStore, and WebHCat. Ensure that the parameter values are the same.

Step 7 Choose **Cluster > Services > Hive > Configuration > All Configurations**, search for **maxConnectionsPerPartition**, and check whether its value is less than **100**.

- If yes, change the value to **100**, click **Save**, and go to **Step 8**.
- If no, go to **Step 8**.

Step 8 If parameters are modified in the preceding steps, choose **Cluster > Services > Hive > Dashboard** and choose **More > Service Rolling Restart**. If the parameters are not modified, skip this step.

----End

16.10.48 WebHCat Failed to Start Due to Abnormal Health Status

Issue

The WebHCat instance fails to be started.

Symptom

On Manager, the health status of the WebHCat instance is **Faulty**, and alarm **ALM-12007 Process Fault** is generated for the WebHCat instance of the Hive service. An error is reported when the Hive service is restarted.

Error messages "Service not found in Kerberos database" and "Address already in use" are contained in the **/var/log/Bigdata/hive/webhcat/webhcat.log** file of the WebHCat instance.

Procedure

Step 1 Log in to each node where the WebHCat instance resides and check whether the mapping between IP addresses and hostnames in the **/etc/hosts** file is correct. The WebHCat configurations in the **/etc/hostname** and **/etc/HOSTNAME** files must be the same as those in the **/etc/hosts** file. If they are different, manually modify them.

NOTE

To view the mapping between the IP addresses and hostnames of the WebHCat instance, log in to FusionInsight Manager and choose **Cluster** > **Services** > **Hive** > **Instance**.

Step 2 Log in to any node where the WebHCat instance resides and run the following command to switch to user **omm**:

su - omm

Step 3 Run the following command to check whether the WebHCat process exists:

ps -ef|grep webhcat|grep -v grep

If it does, run the following command to kill it:

kill -9 ${webhcat_pid}

Step 4 Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Hive** . On the page that is displayed, click the **Instance** tab. The select all WebHCat instances, click **More**, and select **Restart Instance**. Wait until WebHCat is restarted successfully.

----End

16.10.49 WebHCat Failed to Start Because the mapred-default.xml File Cannot Be Parsed

Issue

The Hive service of MRS is faulty. After the Hive service is restarted, the HiveServer and WebHCat processes on the Master2 node fail to start, but the processes on the Master1 node are normal.

Cause Analysis

Log in to the Master2 node and check the **/var/log/Bigdata/hive/hiveserver/hive.log** file. It is found that HiveServer keeps loading **/opt/Bigdata/*/*_HiveServer/etc/hive-site.xml**. Check the **/var/log/Bigdata/hive/hiveserver/hiveserver.out** log generated when HiveServer exits. It is found that an exception occurs when the **mapred-default.xml** file is parsed.

Procedure

Step 1 Log in to the Master2 node and run the following command to query the path of **mapred-default.xml**:

find /opt/ -name 'mapred-default.xml'

The configuration file is in the **/opt/Bigdata/*/*_WebHCat/etc/** directory but is empty.

Step 2 Log in to the Master1 node, copy the **/opt/Bigdata/*/*_WebHCat/etc/mapred-default.xml** file to the Master2 node, and change the owner group of the file to **omm:wheel**.

Step 3 Log in to Manager and restart the abnormal HiveServer and WebHCat instances.

----End

16.11 Using Hue

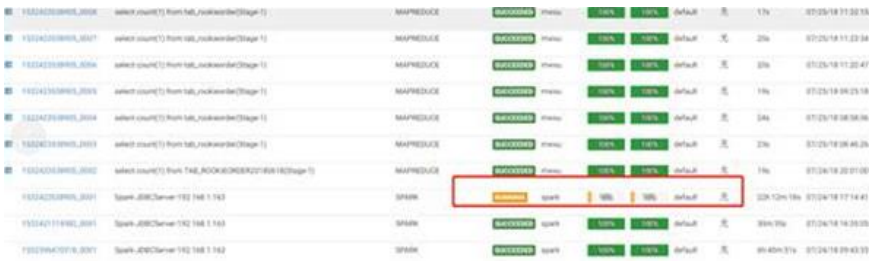
16.11.1 A Job Is Running on Hue

Issue

The customer finds that a job is running on Hue.

Symptom

After the customer's MRS is installed, the job is running on Hue but the running job is not operated by the customer.



Cause Analysis

This job is a permanent job generated when the system connects to JDBC after Spark is started.

Procedure

This is not a problem. No handling is required.

16.11.2 HQL Fails to Be Executed on Hue Using Internet Explorer

Symptom

Using Internet Explorer to access Hive Editor and execute all HQL statements on Hue fails and the system prompts "There was an error with your query".

Cause Analysis

Internet Explorer has functional problems and cannot process AJAX POST requests containing form data in 307 redirection. Use a compatible browser.

Solution

Use Google Chrome 21 or later.

16.11.3 Hue (Active) Cannot Open Web Pages

Symptom

The following information is displayed on the web UI of Hue (active):

Service Unavailable
The server is temporarily unable to service your request due to maintenance downtime or capacity problems. Please try again later.

Cause Analysis

- The Hue configuration has expired.
- The configuration of the Hue service needs to be modified manually in a single-master cluster.

Solution

- If the Hue configuration has expired, restart the Hue service.
- Manually modify the Hue service configuration for a single-master cluster.
  a. Log in to the Master node.
  b. Run the **hostname -i** command to obtain the IP address of the local host.
  c. Run the following command to obtain the value of **HUE_FLOAT_IP**:

     grep "HUE_FLOAT_IP" ${BIGDATA_HOME}/*MRS*_Current/1_*/etc*/ENV_VARS,

     Replace *MRS* with the actual file name.
  d. Check whether the local IP address is the same as the value of **HUE_FLOAT_IP**. If they are different, change the value of **HUE_FLOAT_IP** to the local IP address.
  e. Restart the Hue service.

16.11.4 Failed to Access the Hue Web UI

Issue

An error page is displayed when the Hue web UI is accessed.

Symptom

The following error information is displayed on the Hue web UI:

```
503 Service Unavailable
The server is temporarily unable to service your requster due to maintenance downtime or capacity problems.Please try again later.
```

Cause Analysis

- The Hue configuration has expired.
- The configuration of the Hue service needs to be modified manually in a single-master cluster.

Procedure

Step 1 Log in to the Master node.

Step 2 Run the **hostname -i** command to obtain the IP address of the local host.

Step 3 Run the following command to obtain the value of **HUE_FLOAT_IP**:

grep "HUE_FLOAT_IP" ${BIGDATA_HOME}/*MRS*_Current/1_*/etc*/ENV_VARS, where *MRS* is subject to the actual file name.

Step 4 Check whether the local IP address is the same as the value of **HUE_FLOAT_IP**. If they are different, change the value of **HUE_FLOAT_IP** to the local IP address.

Step 5 Restart the Hue service.

----End

16.11.5 HBase Tables Cannot Be Loaded on the Hue Web UI

Issue

After Hive data is imported to HBase on the Hue page, an error message is displayed, indicating that the HBase table cannot be detected.

Symptom

In the Kerberos cluster, the IAM sub-account does not have sufficient permissions. As a result, the HBase table cannot be loaded.

Cause Analysis

The IAM subaccount does not have sufficient permissions.

Procedure

MRS Manager:

Step 1 Log in to MRS Manager.

Step 2 Choose **System** > **Manage User**.

Step 3 Locate the row that contains the target user, and click **Modify**.

Step 4 Add the user to the **supergroup** group.

Step 5 Click **OK**. The modification is complete.

----End

FusionInsight Manager:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the target user, and click **Modify**.

Step 4 Add the user to the **supergroup** group.

Step 5 Click **OK**. The modification is complete.

----End

Summary and Suggestions

If Kerberos authentication is enabled for a cluster, "No data available" is displayed on the page. In this case, check the permission first.

16.12 Using Impala

16.12.1 Failed to Connect to impala-shell

Issue

A user fails to connect to impala-shell.

Symptom

After a user modifies the configuration of any component on the component management page and restarts the service, the connection to impala-shell fails, and the error message "no such file/directory" is displayed.

```
[root@node-master1emdj etc]# pwd
/opt/Bigdata/MRS_2.1.0/1_7_KuduMaster/etc
[root@node-master1emdj etc]# impala-shell -i 192.168.0.73
shell-init: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory
chdir: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory
Traceback (most recent call last):
  File "/opt/client/Impala/impala/shell/impala_shell.py", line 38, in <module>
    from impala_client import (ImpalaClient, DisconnectedException, QueryStateException,
  File "/opt/client/Impala/impala/shell/lib/impala_client.py", line 20, in <module>
    import sasl
  File "build/bdist.linux-x86_64/egg/sasl/__init__.py", line 1, in <module>

  File "build/bdist.linux-x86_64/egg/sasl/saslwrapper.py", line 7, in <module>
  File "build/bdist.linux-x86_64/egg/_saslwrapper.py", line 7, in <module>
  File "build/bdist.linux-x86_64/egg/_saslwrapper.py", line 3, in __bootstrap__
  File "/usr/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg/pkg_resources.py", line 2594, in <module>
    for comparator, version in req.specs:
  File "/usr/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg/pkg_resources.py", line 425, in __init__

  File "/usr/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg/pkg_resources.py", line 440, in add_entry
    `req`.  But, if there is an active distribution for the project and it
  File "/usr/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg/pkg_resources.py", line 1688, in find_on_path
    return ()
  File "/usr/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg/pkg_resources.py", line 1835, in _normalize_cached

  File "/usr/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg/pkg_resources.py", line 1829, in normalize_path
    register_namespace_handler(object,null_ns_handler)
  File "/usr/lib64/python2.7/posixpath.py", line 368, in realpath
    return abspath(path)
  File "/usr/lib64/python2.7/posixpath.py", line 356, in abspath
    cwd = os.getcwd()
OSError: [Errno 2] No such file or directory
```

Cause Analysis

After the service configuration is modified and the service is restarted, some directory structures of the service, such as the etc directory, are deleted and recreated. If the directory is etc or its subdirectory before the service is restarted, some system variables or parameters cannot be found when impala-shell is executed in the original directory because the directory is recreated after the service is restarted. As a result, impala-shell fails to be connected.

Procedure

Switch to any existing directory and reconnect to impala-shell.

16.12.2 Failed to Create a Kudu Table

Issue

An error occurs when a user creates a Kudu table.

Symptom

When a user creates table in a new cluster, the following error message is displayed: [Cloudera]ImpalaJDBCDriver ERROR processing query/statement. Error

Code: 0, SQL state: TStatus(statusCode:ERROR_STATUS, sqlState:HY000, errorMessage:AnalysisException: Table property 'kudu.master_addresses' is required when the impalad startup flag -kudu_master_hosts is not used.

Cause Analysis

The user does not specify **kudu.master_addresses** in the Impala SQL statement.

Procedure

Specify **kudu.master_addresses** when creating a Kudu table.

16.12.3 Failed to Log In to the Impala Client

Issue

Error information similar to the following is displayed when a user runs the Impala client.

```
[root@node-master1avIy ~]# impala-shell -i 192.168.128.49:21000
  File "/opt/client/Impala/impala/shell/impala_shell.py", line 1675
    except Exception, e:
                    ^
SyntaxError: invalid syntax
[root@node-master1avIy ~]#
```

Cause Analysis

The latest MRS cluster uses EulerOS 2.9 or later, which provides only Python 3. However, the Impala client is implemented based on Python 2 and is incompatible with some syntax of Python 3. As a result, an error occurs when the Impala client is running. You can manually install Python 2 to solve this problem.

Procedure

Step 1 Log in to the Impala node as user **root** and run the following command to check its Python version:

python --version

```
[root@node-master2JgOY ~]# python --version
Python 3.7.4
```

Step 2 Run the **yum install make** command to check whether yum is available.

- If the following error is reported, the yum configuration is incorrect. Go to **Step 3**.

  ```
  [root@node-master2JgOY ~]# yum install make
  Error: There are no enabled repositories in "/etc/yum.repos.d", "/etc/yum/repos.d", "/etc/distro.repos.d".
  ```

- If no error is reported, go to **Step 4**.

Step 3 Run the **cat /etc/yum.repos.d/EulerOS-base.repo** command to check whether the yum source matches the system version. If they do not match, modify them.

Before modification

```
[root@node-master1avIy ~]# cat /etc/yum.repos.d/EulerOS-base.repo
[base]
name=EulerOS-2.0SP2 base
baseurl=http://mirrors.myhuaweicloud.com/euler/ict/site-euleros/euleros/repo/yum/2.2/os/x86_64/
enabled=1
gpgcheck=1
gpgkey=http://mirrors.myhuaweicloud.com/euler/ict/site-euleros/euleros/repo/yum/2.2/os/RPM-GPG-KEY-EulerOS
[root@node-master1avIy ~]# uname -a
Linux node-master1avIy.mrs-mq7v.com 4.18.0-147.5.1.6.h541.eulerosv2r9.x86_64 #1 SMP Wed Aug 4 02:30:13 UTC
 x86_64 GNU/Linux
```

After modification

```
[base]
name=EulerOS-2.0SP9 base
baseurl=http://mirrors.myhuaweicloud.com/euler/ict/site-euleros/euleros/repo/yum/2.9/os/x86_64/
enabled=1
gpgcheck=1
gpgkey=http://mirrors.myhuaweicloud.com/euler/ict/site-euleros/euleros/repo/yum/2.9/os/RPM-GPG-KEY-EulerOS
```

Step 4 Run the following command to check for the software whose name starts with **python2** in the yum source:

yum list python2*

```
[root@node-master2JgOY ~]# yum list python2*
Last metadata expiration check: 0:02:36 ago on Thu 16 Dec 2021 10:05:52 AM CST.
Available Packages
python2.x86_64                                  2.7.16-16.eulerosv2r9
python2-debug.x86_64                            2.7.16-16.eulerosv2r9
python2-devel.x86_64                            2.7.16-16.eulerosv2r9
python2-help.noarch                             2.7.16-16.eulerosv2r9
python2-pip.noarch                              18.0-13.h2.eulerosv2r9
python2-setuptools.noarch                       40.4.3-4.h1.eulerosv2r9
python2-tkinter.x86_64                          2.7.16-16.eulerosv2r9
python2-tools.x86_64                            2.7.16-16.eulerosv2r9
```

Step 5 Run the following command to install Python 2:

yum install python2

```
[root@node-master2JgOY ~]# yum install python2
Last metadata expiration check: 0:00:48 ago on Thu 16 Dec 2021 10:05:52 AM CST.
Error:
 Problem: problem with installed package python3-unversioned-command-3.7.4-7.h29.eulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h29.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h11.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h13.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h15.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h18.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h33.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - package python3-unversioned-command-3.7.4-7.h38.eulerosv2r9.x86_64 conflicts with python2 provided by python2-2.7.16-16.e
ulerosv2r9.x86_64
  - conflicting requests
(try to add '--allowerasing' to command line to replace conflicting packages or '--skip-broken' to skip uninstallable package
s or '--nobest' to use not only best candidate packages)
```

Python 3 has been installed in the current system. If you directly install Python 2, a conflict message is displayed.

You can select **--allowerasing** or **--skip-broken** for the installation. For example:

yum install python2 --skip-broken

```
[root@node-master2JgOY ~]# yum install python2 --skip-broken
Last metadata expiration check: 0:34:08 ago on Thu 16 Dec 2021 10:05:52 AM CST.
Dependencies resolved.
================================================================================================================================
 Package                          Architecture         Version                                Repository          Size
================================================================================================================================
Installing:
 python2                          x86_64               2.7.16-16.eulerosv2r9                  base               6.4 M
Installing dependencies:
 libXft                           x86_64               2.3.2-13.eulerosv2r9                   base                41 k
```

After the installation is complete, the Python version is automatically changed to python2, as shown in the following figure.

```
Installed:
  libXft-2.3.2-13.eulerosv2r9.x86_64                          libXrender-0.9.10-10.eulerosv2r9.x86_64
  python2-2.7.16-16.eulerosv2r9.x86_64                        python2-debug-2.7.16-16.eulerosv2r9.x86_64
  python2-devel-2.7.16-16.eulerosv2r9.x86_64                  python2-help-2.7.16-16.eulerosv2r9.noarch
  python2-setuptools-40.4.3-4.h1.eulerosv2r9.noarch           python2-tkinter-2.7.16-16.eulerosv2r9.x86_64
  python2-tools-2.7.16-16.eulerosv2r9.x86_64                  python3-rpm-generators-9-1.eulerosv2r9.noarch
  tk-1:8.6.8-5.eulerosv2r9.x86_64

Complete!
[root@node-master2JgOY ~]# python --version
Python 2.7.16
```

If Python 2 is installed successfully but the displayed Python version is incorrect, run the following command to create the **/usr/bin/python** soft link for **/usr/bin/python2**:

ln -s /usr/bin/python2 /usr/bin/python

Step 6 Verify that the Impala client is available.

```
[root@node-master1avIy ~]# impala-shell -i 192.168.128.49:21000
Starting Impala Shell without Kerberos authentication
Opened TCP connection to 192.168.128.49:21000
Connected to 192.168.128.49:21000
Server version: impalad version 3.4.0-RELEASE RELEASE (build eebadd34c1563cbf5825a4e4d361e7b3601f9827)
***********************************************************************************
Welcome to the Impala shell.
(Impala Shell v3.4.0-RELEASE (eebadd3) built on Thu Nov  4 11:29:54 CST 2021)

After running a query, type SUMMARY to see a summary of where time was spent.
***********************************************************************************
[192.168.128.49:21000] default> show databases;
Query: show databases
+------------------+----------------------------------------------+
| name             | comment                                      |
+------------------+----------------------------------------------+
| _impala_builtins | System database for Impala builtin functions |
| default          | Default Hive database                        |
+------------------+----------------------------------------------+
Fetched 2 row(s) in 0.16s
[192.168.128.49:21000] default>
```

----End

16.13 Using Kafka

16.13.1 An Error Is Reported When Kafka Is Run to Obtain a Topic

Issue

An Error is reported when Kafka is run to obtain a topic.

Symptom

An error is reported when the Kafka is run to obtain topics. The error information is as follows:

```
ERROR org.apache.kafka.common.errors.InvalidReplicationFactorException: Replication factor: 2 larger than available brokers: 0.
```

Possible Cause

The variable for obtaining the ZooKeeper address is incorrect due to special characters.

Procedure

Step 1 Log in to any Master node.

Step 2 Run the **cat /opt/client/Kafka/kafka/config/server.properties |grep '^zookeeper.connect ='** command to check the variable of the Zookeeper address.

Step 3 Run Kafka again to obtain the topic. Do not add any character to the variables obtained in **Step 2**.

----End

16.13.2 Flume Normally Connects to Kafka But Fails to Send Messages

Symptom

An MRS cluster is installed, and ZooKeeper, Flume, and Kafka are installed in the cluster.

Flume fails to send data to Kafka.

Possible Causes

1. The Kafka service is abnormal.
2. The IP address for Flume to connect to Kafka is incorrect.
3. The size of the message sent from Flume to Kafka exceeds the upper limit.

Cause Analysis

The possible reasons why Flume fails to send data to Kafka may be related to Flume or Kafka.

1. Check the Kafka service status and monitoring metrics on Manager.
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**.

     Check the Kafka status. It is found that the status is good and the monitoring metrics are correctly displayed.

2. Check the Flume log. The log contains **MessageSizeTooLargeException** information, as shown in the following:

```
2016-02-26 14:55:19,126 | WARN  | [SinkRunner-PollingRunner-DefaultSinkProcessor] |  Produce request with correlation id 349829 failed due to [LOG,7]: kafka.common.MessageSizeTooLargeException  | kafka.utils.Logging$class.warn(Logging.scala:83)
```

   The exception shows that the size of data written to Kafka by Flume exceeds the maximum message size specified by Kafka.

3. Check the maximum message size specified by Kafka on Manager.
   - MRS Manager portal: Log in to MRS Manager and choose **Services** > **Kafka** > **Configuration**.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka** > **Configuration**.

   On the page that is displayed, set **Type** to **All**. All Kafka configurations are displayed. Enter **message.max.bytes** in the **Search** text box to search.

   In MRS, the maximum size of a message that can be received by the Kafka server is 1000012 bytes = 977 KB by default.

Solution

After confirmation with the customer, data sent by Flume contains messages over 1 MB. Adjust parameters on Kafka to enable the messages to be written to Kafka.

Step 1 Set **message.max.bytes** to a value that is larger than the current maximum size of the message to be written so that Kafka can receive all messages.

Step 2 Set **replica.fetch.max.bytes** to a value that is equal to or larger than the value of **message.max.bytes** so that replicas of partitions on different Brokers can be synchronized to all messages.

- MRS Manager portal: Log in to MRS Manager and choose **Services** > **Kafka** > **Configuration**.
- FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka** > **Configuration**.

On the page that is displayed, set **Type** to **All**. All Kafka configurations are displayed. Enter **replica.fetch.max.bytes** in the **Search** text box to search.

Step 3 Click **Save** and restart the Kafka service to make Kafka configurations take effect.

Step 4 Set **fetch.message.max.bytes** to a value that is equal to or larger than the value of **message.max.bytes** for Consumer service applications to ensure that Consumers can consume all messages.

----End

16.13.3 Producer Failed to Send Data and Threw "NullPointerException"

Symptom

An MRS cluster has ZooKeeper and Kafka installed.

When the Producer client sends data to Kafka, it fails and throws "NullPointerException".

Possible Causes

1. The Kafka service is abnormal.
2. The **jass** and **keytab** files configured on the Producer client are incorrect.

Cause Analysis

The possible causes may be related to Producer or Kafka.

1. Check the Kafka service status and monitoring metrics on Manager.
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster**. Click the name of the target cluster and choose **Service** > **Kafka**.

     Check the Kafka status. The status is good, and the monitoring metrics are correctly displayed.
2. Check the Producer client log. The log contains "NullPointerException", as shown in **Figure 16-42**.

   Figure 16-42 Producer client log

   ```
   [2016-12-06 02:04:05,906]-[schedule-C50D0717-4643-4D4E-9D6E-B940E4BD7159]-[kafka-producer-network-thread |
   SZX1000161910-10.21.219.222-bigdata-producer-5]-[1005]-[org.apache.kafka.clients.producer.internals.Sender.run
   thread:
   java.lang.NullPointerException
       at org.apache.kafka.common.network.Selector.pollSelectionKeys(Selector.java:302)
       at org.apache.kafka.common.network.Selector.poll(Selector.java:283)
       at org.apache.kafka.clients.NetworkClient.poll(NetworkClient.java:260)
       at org.apache.kafka.clients.producer.internals.Sender.run(Sender.java:229)
       at org.apache.kafka.clients.producer.internals.Sender.run(Sender.java:134)
       at java.lang.Thread.run(Thread.java:745)
   [2016-12-06 02:04:05,921]-[schedule-C50D0717-4643-4D4E-9D6E-B940E4BD7159]-[kafka-producer-network-thread |
   SZX1000161910-10.21.219.222-bigdata-producer-3]-[1005]-[org.apache.kafka.clients.producer.internals.Sender.run
   thread:
   java.lang.NullPointerException
       at org.apache.kafka.common.network.Selector.pollSelectionKeys(Selector.java:302)
       at org.apache.kafka.common.network.Selector.poll(Selector.java:283)
       at org.apache.kafka.clients.NetworkClient.poll(NetworkClient.java:260)
       at org.apache.kafka.clients.producer.internals.Sender.run(Sender.java:229)
       at org.apache.kafka.clients.producer.internals.Sender.run(Sender.java:134)
       at java.lang.Thread.run(Thread.java:745)
   ```

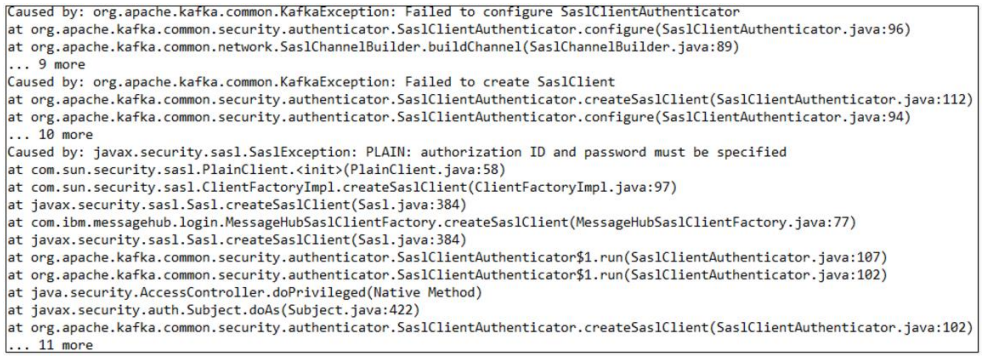
   Alternatively, the log contains only "NullPointerException" but no stack information. The problem is caused by JDK self-protection. If much information is printed for the same stack, the JDK self-protection is triggered and stack information is no longer printed, as shown in **Figure 16-43**.

   Figure 16-43 Error information

   ```
   [2016-11-23 04:06:53,973] [kafka-producer-network-thread | producer-1] [ERROR] [org.apache.kafka.clients.producer.internals.Sender] (run:130)- Uncaught error in kafka producer I/O thread:
   java.lang.NullPointerException
   [2016-11-23 04:06:53,973] [kafka-producer-network-thread | producer-1] [ERROR] [org.apache.kafka.clients.producer.internals.Sender] (run:130)- Uncaught error in kafka producer I/O thread:
   java.lang.NullPointerException
   [2016-11-23 04:06:53,973] [kafka-producer-network-thread | producer-1] [ERROR] [org.apache.kafka.clients.producer.internals.Sender] (run:130)- Uncaught error in kafka producer I/O thread:
   java.lang.NullPointerException
   [2016-11-23 04:06:53,973] [kafka-producer-network-thread | producer-1] [ERROR] [org.apache.kafka.clients.producer.internals.Sender] (run:130)- Uncaught error in kafka producer I/O thread:
   java.lang.NullPointerException
   ```

3. Check the Producer client log. Error information "Failed to configure SaslClientAuthenticator" is displayed, as shown in **Figure 16-44**.

Figure 16-44 Error log

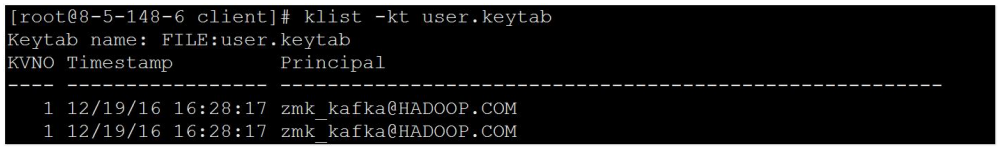


4. The authentication failure causes the failure to create the KafkaChannel. The KafkaChannel obtained through the **channel(key)** method is empty and "NullPointerException" is excessively printed. According to the preceding log, the authentication fails due to an incorrect password which does not match the username.
5. Check the **jaas** and **keytab** files. The **principal** is set to **stream** in the **jaas** file.

Figure 16-45 Checking the jaas file

The **principal** is set to **zmk_kafka** in the **user.keytab** file.

Figure 16-46 Checking the user.keytab file



The **principal** in the **jaas** file is inconsistent with that in the **user.keytab** file.

The application automatically and periodically updates the **jaas** file. However, when two processes of the application update the **jaas** file, one process writes a correct **principal** whereas the other process writes an incorrect one. As a result, the application is abnormal sometimes.

Procedure

Step 1 Modify the **jaas** file to ensure that its **principal** exists in the **keytab** file.

----End

16.13.4 Producer Fails to Send Data and "TOPIC_AUTHORIZATION_FAILED" Is Thrown

Symptom

An MRS cluster is installed, and ZooKeeper and Kafka are installed in the cluster.

When Producer sends data to Kafka, the client throws "TOPIC_AUTHORIZATION_FAILED."

Possible Causes

1. The Kafka service is abnormal.
2. The Producer client adopts non-security access and access is disabled on the server.
3. The Producer client adopts non-security access and ACL is set for Kafka topics.

Cause Analysis

The possible reasons why Producer fails to send data to Kafka may be related to Producer or Kafka.

1. Check the Kafka service status:
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**.

     Check the Kafka status. It is found that the status is good and the monitoring metrics are correctly displayed.
2. Check the Producer client logs. The logs contain the error information "TOPIC_AUTHORIZATION_FAILED."

   ```
   [root@10-10-144-2 client]# kafka-console-producer.sh --broker-list 10.5.144.2:9092 --topic test
   1
   [2017-01-24 16:58:36,671] WARN Error while fetching metadata with correlation id 0 :
   {test=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkClient)
   [2017-01-24 16:58:36,672] ERROR Error when sending message to topic test with key: null, value: 1
   bytes with error: Not authorized to access topics: [test]
   (org.apache.kafka.clients.producer.internals.ErrorLoggingCallback)
   ```

   Producer accesses Kafka using port 9092, which is a non-security port.
3. On Manager, check the current Kafka cluster configuration. It is found that the customized configuration **allow.everyone.if.no.acl.found**=**false** is not configured.
   - MRS Manager portal: Log in to MRS Manager and choose **Services** > **Kafka** > **Configuration**.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka** > **Configuration**.
4. If ACL is set to **false**, port 9092 cannot be used for access.
5. Check the Producer client logs. The logs contain the error information "TOPIC_AUTHORIZATION_FAILED."

   ```
   [root@10-10-144-2 client]# kafka-console-producer.sh --broker-list 10.5.144.2:21005 --topic test_acl
   1
   ```

```
[2017-01-25 11:09:40,012] WARN Error while fetching metadata with correlation id 0 :
{test_acl=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkClient)
[2017-01-25 11:09:40,013] ERROR Error when sending message to topic test_acl with key: null, value:
1 bytes with error: Not authorized to access topics: [test_acl]
(org.apache.kafka.clients.producer.internals.ErrorLoggingCallback)
[2017-01-25 11:14:40,010] WARN Error while fetching metadata with correlation id 1 :
{test_acl=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkClient)
```

Producer accesses Kafka using port 21005, which is a non-security port.

6. Run the client command to check the ACL permission of the topic.

```
[root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:24002/
kafka --list --topic topic_acl
Current ACLs for resource `Topic:topic_acl`:
 User:test_user has Allow permission for operations: Describe from hosts: *
User:test_user has Allow permission for operations: Write from hosts: *
```

If ACL is set for the topic, port 9092 cannot be used for access.

7. Check the Producer client logs. The logs contain the error information "TOPIC_AUTHORIZATION_FAILED."

```
[root@10-10-144-2 client]# kafka-console-producer.sh --broker-list 10.5.144.2:21007 --topic topic_acl
--producer.config /opt/client/Kafka/kafka/config/producer.properties
1
[2017-01-25 12:43:58,506] WARN Error while fetching metadata with correlation id 0 :
{topic_acl=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkClient)
[2017-01-25 12:43:58,507] ERROR Error when sending message to topic topic_acl with key: null,
value: 1 bytes with error: Not authorized to access topics: [topic_acl]
(org.apache.kafka.clients.producer.internals.ErrorLoggingCallback)
```

Producer uses port 21007 to access Kafka.

8. Run the client command **klist** to query the current authenticated user.

```
[root@10-10-144-2 client]# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: test@HADOOP.COM

Valid starting     Expires            Service principal
01/25/17 11:06:48  01/26/17 11:06:45  krbtgt/HADOOP.COM@HADOOP.COM
```

The **test** user is used in this example.

9. Run the client command to check the ACL permission of the topic.

```
[root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/
kafka --list --topic topic_acl
Current ACLs for resource `Topic:topic_acl`:
 User:test_user has Allow permission for operations: Describe from hosts: *
User:test_user has Allow permission for operations: Write from hosts: *
```

After ACL is set for the topic, user **test_user** has Producer permission. User **test** has no permission to perform Producer operations.

For details about the solution, see **2**.

10. Log in to Kafka Broker using SSH.

Run the **cd /var/log/Bigdata/kafka/broker** command to go to the log directory.

Check the **kafka-authorizer.log** file. It shows that the user does not belong to the **kafka** or **kafkaadmin** group.

```
2017-01-25 13:26:33,648 | INFO  | [kafka-request-handler-0] | The principal is test, belongs to Group :
[hadoop, ficommon] | kafka.authorizer.logger (SimpleAclAuthorizer.scala:169)
2017-01-25 13:26:33,648 | WARN  | [kafka-request-handler-0] | The user is not belongs to kafka or
kafkaadmin group, authorize failed! | kafka.authorizer.logger (SimpleAclAuthorizer.scala:170)
```

For details about the solution, see **3**.

Solution

Step 1 Set **allow.everyone.if.no.acl.found** to **true** and restart the Kafka service.

Step 2 Use the account with permission for login.

Example:

kinit test_user

Alternatively, grant the user with related permission.

NOTICE

This operation must be performed by the Kafka administrator (belonging to the **kafkaadmin** group).

Example:

kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/kafka --topic topic_acl --producer --add --allow-principal User:test

```
[root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=8.5.144.2:2181/kafka --list --topic topic_acl
Current ACLs for resource `Topic:topic_acl`:
 User:test_user has Allow permission for operations: Describe from hosts: *
 User:test_user has Allow permission for operations: Write from hosts: *
  User:test has Allow permission for operations: Describe from hosts: *
  User:test has Allow permission for operations: Write from hosts: *
```

Step 3 Add the user to the **kafka** or **kafkaadmin** group.

----End

16.13.5 Producer Occasionally Fails to Send Data and the Log Displays "Too many open files in system"

Symptom

When Producer sends data to Kafka, it is found that the client fails to send data.

Figure 16-47 Producer fails to send data.

Possible Causes

1. The Kafka service is abnormal.
2. The network is abnormal.
3. The Kafka topic is abnormal.

Cause Analysis

1. Check the Kafka service status:
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**.

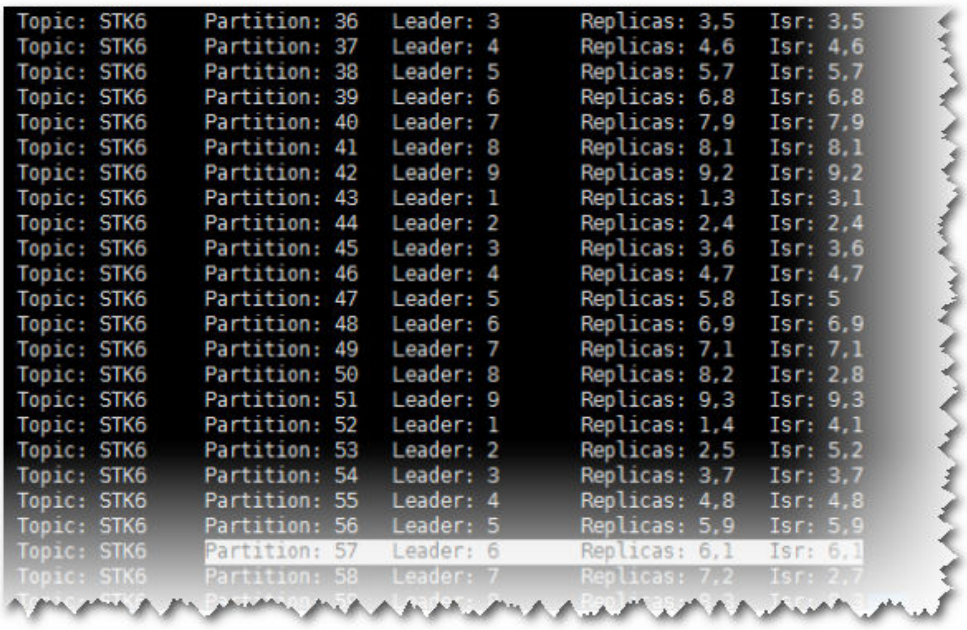
     Check the Kafka status. It is found that the status is good and the monitoring metrics are correctly displayed.
2. View the error topic information in the SparkStreaming log.

   Run the Kafka commands to obtain the topic assignment information and copy synchronization information, and check the return result.

   kafka-topics.sh --describe --zookeeper <zk_host:port/chroot>

   As shown in **Figure 16-48**, the topic status is normal. All partitions have normal leader information.

   Figure 16-48 Topic status

3. Run the **telnet** command to check whether the Kafka can be connected.

   telnet *Kafka service IP address Kafka service port*

   If telnet fails, check the network security group and ACL.
4. Log in to Kafka Broker using SSH.

   Run the **cd /var/log/Bigdata/kafka/broker** command to go to the log directory.

   Check on **server.log** indicates that the error message is displayed in the log shown in the following figure.

Figure 16-49 Log exception

5. Output of the **lsof** command used to check the handle usage of the Kafka process on the current node shows that the number of handles used by the Kafka process reaches 470,000.

Figure 16-50 Handles

6. Check the service code. It is found that the Producer object is frequently created and is not closed normally.

Solution

Step 1 Stop the current application to ensure that the number of handles on the server does not increase sharply, which affects the normal running of services.

Step 2 Optimize the application code to resolve the handle leakage problem.

Suggestion: Use one Producer object globally. After the use is complete, call the Close interface to close the handle.

----End

16.13.6 Consumer Is Initialized Successfully, But the Specified Topic Message Cannot Be Obtained from Kafka

Symptom

An MRS cluster is installed, and ZooKeeper, Flume, Kafka, Storm, and Spark are installed in the cluster.

The customer cannot consume any data using Storm, Spark, Flume or self-programmed Consumer code to consume messages of the specified Kafka topic.

Possible Causes

1. The Kafka service is abnormal.
2. The IP address for ZooKeeper connection is incorrectly set.
3. "ConsumerRebalanceFailedException" is thrown.

4. "ClosedChannelException" caused by network problems is thrown.

Cause Analysis

Storm, Spark, Flume or user-defined Consumer code can be called Consumer.

1. Check the Kafka service status:
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**.

     Check the Kafka status. It is found that the status is good and the monitoring metrics are correctly displayed.
2. Check whether data can be normally consumed through the Kafka client.

   Suppose the client has been installed in the **/opt/client** directory, **test** is the topic name to be consumed, and the IP address of ZooKeeper is 192.168.234.231.

   ```
   cd /opt/client
   source bigdata_env
   kinit admin
   kafka-topics.sh --zookeeper 192.168.234.231:2181/kafka --describe --topic testkafka-console-consumer.sh --topic test --zookeeper 192.168.234.231:2181/kafka --from-beginning
   ```

   If data can be consumed, the cluster service is running properly.
3. Check Consumer configurations. The IP address for connecting to ZooKeeper is incorrect.
   - Flume

     ```
     server.sources.Source02.type=org.apache.flume.source.kafka.KafkaSource

     server.sources.Source02.zookeeperConnect=192.168.234.231:2181
     server.sources.Source02.topic = test
     server.sources.Source02.groupId = test_01
     ```
   - Spark

     ```
     val zkQuorum = "192.168.234.231:2181"
     ```
   - Storm

     ```
     BrokerHosts brokerHosts = new ZKHosts("192.168.234.231:2181");
     ```
   - Consumer API

     ```
     zookeeper.connect="192.168.234.231:2181"
     ```

   On MRS Manager, the root path of ZNode where Kafka is stored on ZooKeeper is **/kafka**, which is differentiated from the open source. The address for Kafka to connect to ZooKeeper is **192.168.234.231:2181/kafka**.

   However, the address for Consumer to connect to ZooKeeper is **192.168.234.231:2181**. Therefore, topic information about Kafka cannot be correctly obtained.

   For details about the solution, see **Step 1**.
4. Check Consumer logs. The logs contain "ConsumerRebalanceFailedException".

   ```
   2016-02-03 15:55:32,557 | ERROR | [ZkClient-EventThread-75- 192.168.234.231:2181/kafka] |  Error handling event ZkEvent[New session event sent to kafka.consumer.ZookeeperConsumerConnector$ZKSessionExpireListener@34b41dfe]  | org.I0Itec.zkclient.ZkEventThread.run(ZkEventThread.java:77)
   kafka.common.ConsumerRebalanceFailedException: pc-zjqbetl86-1454482884879-2ec95ed3 can't rebalance after 4 retries
   at kafka.consumer.ZookeeperConsumerConnector$ZKRebalancerListener.syncedRebalance(ZookeeperConsumerConnector.scala:633)
   ```

```
at kafka.consumer.ZookeeperConsumerConnector
$ZKSessionExpireListener.handleNewSession(ZookeeperConsumerConnector.scala:487)
at org.I0Itec.zkclient.ZkClient$4.run(ZkClient.java:472)
at org.I0Itec.zkclient.ZkEventThread.run(ZkEventThread.java:71)
```

The exception shows that the current Consumer does not complete rebalance within the specified retry times. As a result, Kafka Topic-Partition is not allocated to Consumer and Consumer cannot consume messages.

For details about the solution, see **Step 3**.

5. Check Consumer logs. The logs contain "Fetching topic metadata with correlation id 0 for topics [Set(test)] from broker [id:26,host:192-168-234-231,port:9092] failed" and "ClosedChannelException".

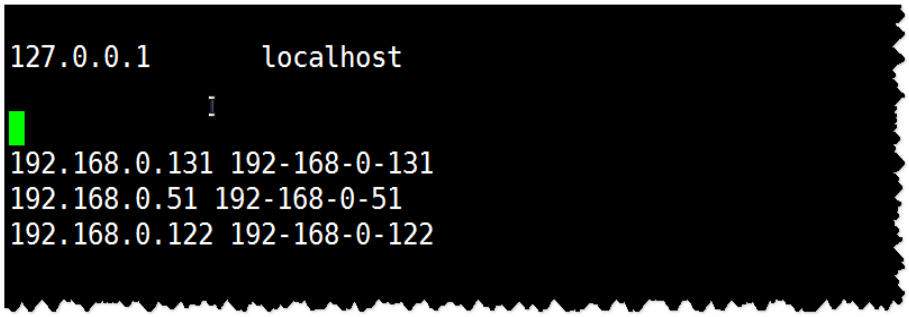
```
[2016-03-04 03:33:53,047] INFO Fetching metadata from broker id:26,host:
192-168-234-231,port:9092 with correlation id 0 for 1 topic(s) Set(test) (kafka.client.ClientUtils$)
[2016-03-04 03:33:55,614] INFO Connected to 192-168-234-231:21005 for producing
(kafka.producer.SyncProducer)
[2016-03-04 03:33:55,614] INFO Disconnecting from 192-168-234-231:21005
(kafka.producer.SyncProducer)
[2016-03-04 03:33:55,615] WARN Fetching topic metadata with correlation id 0 for topics [Set(test)]
from broker [id:26,host: 192-168-234-231,port:21005] failed (kafka.client.ClientUtils$)
java.nio.channels.ClosedChannelException
at kafka.network.BlockingChannel.send(BlockingChannel.scala:100)
at kafka.producer.SyncProducer.liftedTree1$1(SyncProducer.scala:73)
at kafka.producer.SyncProducer.kafka$producer$SyncProducer$$doSend(SyncProducer.scala:72)
at kafka.producer.SyncProducer.send(SyncProducer.scala:113)
at kafka.client.ClientUtils$.fetchTopicMetadata(ClientUtils.scala:58)
at kafka.client.ClientUtils$.fetchTopicMetadata(ClientUtils.scala:93)
at kafka.consumer.ConsumerFetcherManager
$LeaderFinderThread.doWork(ConsumerFetcherManager.scala:66)
at kafka.utils.ShutdownableThread.run(ShutdownableThread.scala:60)
[2016-03-04 03:33:55,615] INFO Disconnecting from 192-168-234-231:21005
(kafka.producer.SyncProducer)
```

The exception shows that the current Consumer cannot obtain metadata from the Kafka Broker 192-168-234-231 node and cannot connect to the correct Broker for obtaining messages.

6. Check the network conditions. If the network is normal, check whether mapping between the host and the IP address is configured.
   - Linux

     Run the **cat /etc/hosts** command.

     Figure 16-51 Example 1

   - Windows

     Open **C:\Windows\System32\drivers\etc\hosts**.

Figure 16-52 Example 2

```
# For example:
#
#      192.168.94.97     rhino.acme.com          # source server
#      192.168.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#       127.0.0.1       localhost
#       ::1             localhost
10.82.129.120 rms.huawei.com # modified by IrmTool at 2015-01-18 17:55:13
```

For details about the solution, see **Step 4**.

Solution

Step 1 The IP address for connecting to ZooKeeper is incorrectly configured.

Step 2 Change the IP address for connecting to ZooKeeper in the Consumer configuration and make it consistent with MRS configuration.

- Flume
  ```
  server.sources.Source02.type=org.apache.flume.source.kafka.KafkaSource
  server.sources.Source02.zookeeperConnect=192.168.234.231:2181/kafka
  server.sources.Source02.topic = test
  server.sources.Source02.groupId = test_01
  ```
- Spark
  ```
  val zkQuorum = "192.168.234.231:2181/kafka"
  ```
- Storm
  ```
  BrokerHosts brokerHosts = new ZKHosts("192.168.234.231:2181/kafka");
  ```
- Consumer API
  ```
  zookeeper.connect="192.168.234.231:2181/kafka"
  ```

Step 3 Rebalance is abnormal.

Multiple Consumers in the same consumer group are successively started and consume data of multiple partitions at the same time, load balancing is performed for Consumers when consumers are fewer than partitions.

The temporary node where the Consumer is stored on ZooKeeper determines read/write permission of which partition of which topic the Consumer has. The path is **/consumers/consumer-group-xxx/owners/topic-xxx/x**.

After the load balancing is triggered, the original Consumer will be recalculated and release occupied partitions, which takes a while. Therefore, new Consumers may fail to preempt the partitions.

Table 16-3 Parameters

Name	Function	Default Value
rebalance.max.retries	Maximum number of rebalance retries	4
rebalance.backoff.ms	Interval for each rebalance retry	2000

Name	Function	Default Value
zookeeper.session.timeout.ms	Maximum time allowed to create a session with ZooKeeper	15000

Set the preceding parameters to higher values. The following is for your reference:

```
zookeeper.session.timeout.ms = 45000
rebalance.max.retries = 10
rebalance.backoff.ms = 5000
```

Parameter setting must comply with the following rule:

rebalance.max.retries * rebalance.backoff.ms > zookeeper.session.timeout.ms

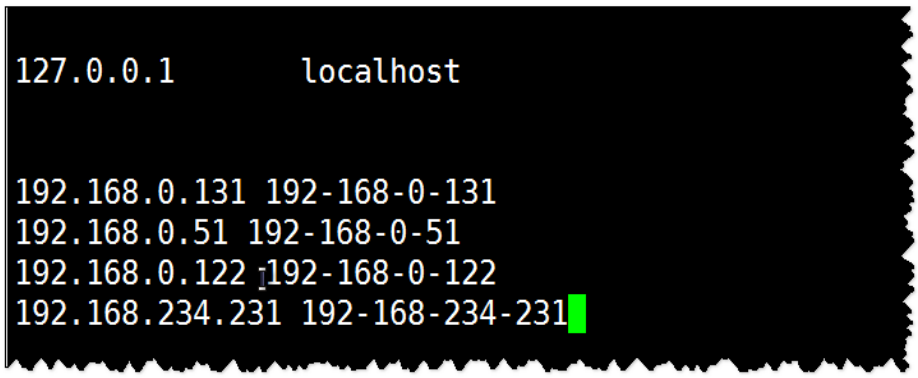
Step 4 The network is abnormal.

In the **hosts** file, mapping between the hostname and IP address is not configured. As a result, information cannot be obtained when using the hostname for access.

Step 5 Add the hostname to the **hosts** file and make it correspond to the IP address.

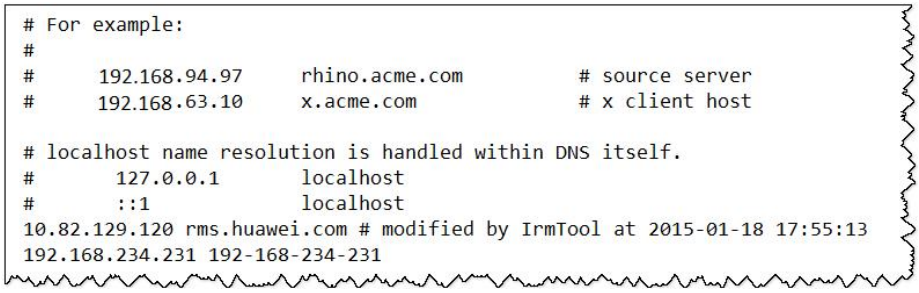
- Linux

  Figure 16-53 Example 3

- Windows

  Figure 16-54 Example 4

----End

16.13.7 Consumer Fails to Consume Data and Remains in the Waiting State

Symptom

An MRS cluster is installed, and ZooKeeper and Kafka are installed in the cluster.

When the Consumer consumes data from Kafka, the client stays in the Waiting state.

Possible Causes

1. The Kafka service is abnormal.
2. The Consumer client adopts non-security access and access is disabled on the server.
3. The Consumer client adopts non-security access and ACL is set for Kafka topics.

Cause Analysis

The possible reasons why the Consumer fails to consume data from Kafka may be related to the Consumer or Kafka.

1. Check the Kafka service status:
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**.

     Check the Kafka status. It is found that the status is good and the monitoring metrics are correctly displayed.
2. Check the Consumer client log. It is found that the information about the frequent connections and disconnections to the Broker node is printed, as shown in the following output.

   ```
   [root@10-10-144-2 client]# kafka-console-consumer.sh --topic test --zookeeper 10.5.144.2:2181/kafka
   --from-beginning



   [2017-03-07 09:22:00,658] INFO Fetching metadata from broker BrokerEndPoint(1,10.5.144.2,9092)
   with correlation id 26 for 1 topic(s) Set(test) (kafka.client.ClientUtils$)
   [2017-03-07 09:22:00,659] INFO Connected to 10.5.144.2:9092 for producing
   (kafka.producer.SyncProducer)
   [2017-03-07 09:22:00,659] INFO Disconnecting from 10.5.144.2:9092 (kafka.producer.SyncProducer)
   ```

   Consumer accesses Kafka using port 9092, which is a non-security port.
3. On Manager, check the current Kafka cluster configuration. It is found that the customized configuration **allow.everyone.if.no.acl.found**=**false** is not configured.
   - MRS Manager portal: Log in to MRS Manager and choose **Services** > **Kafka** > **Configuration**.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka** > **Configuration**.

4. If ACL is set to **false**, port 9092 cannot be used for access.
5. Check the Consumer client log. It is found that the information about the frequent connections and disconnections to the Broker node is printed, as shown in the following output.

```
[root@10-10-144-2 client]# kafka-console-consumer.sh --topic test_acl --zookeeper 10.5.144.2:2181/kafka --from-beginning


[2017-03-07 09:49:16,992] INFO Fetching metadata from broker BrokerEndPoint(2,10.5.144.3,9092) with correlation id 16 for 1 topic(s) Set(topic_acl) (kafka.client.ClientUtils$)
[2017-03-07 09:49:16,993] INFO Connected to 10.5.144.3:9092 for producing (kafka.producer.SyncProducer)
[2017-03-07 09:49:16,994] INFO Disconnecting from 10.5.144.3:9092 (kafka.producer.SyncProducer)
```

   The Consumer accesses Kafka using port 21005, which is a non-security port.
6. Run the client command to check the ACL permission of the topic.

```
[root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/kafka --list --topic topic_acl
Current ACLs for resource `Topic:topic_acl`:
 User:test_user has Allow permission for operations: Describe from hosts: *
User:test_user has Allow permission for operations: Write from hosts: *
```

   If ACL is set for the topic, port 9092 cannot be used for access.
7. The following information is printed in the Consumer client log:

```
[root@10-10-144-2 client]# kafka-console-consumer.sh --topic topic_acl --bootstrap-server 10.5.144.2:21007 --consumer.config /opt/client/Kafka/kafka/config/consumer.properties --from-beginning --new-consumer

[2017-03-07 10:19:18,478] INFO Kafka version : 0.9.0.0 (org.apache.kafka.common.utils.AppInfoParser)
[2017-03-07 10:19:18,478] INFO Kafka commitId : unknown (org.apache.kafka.common.utils.AppInfoParser)
```

   The Consumer uses port 21007 to access Kafka.
8. Run the client command **klist** to query the current authenticated user.

```
[root@10-10-144-2 client]# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: test@HADOOP.COM

Valid starting     Expires            Service principal
01/25/17 11:06:48  01/26/17 11:06:45  krbtgt/HADOOP.COM@HADOOP.COM
```

   The **test** user is used in this example.
9. Run the client command to check the ACL permission of the topic.

```
[root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:24002/kafka --list --topic topic_acl
Current ACLs for resource `Topic:topic_acl`:
 User:test_user has Allow permission for operations: Describe from hosts: *
 User:test_user has Allow permission for operations: Write from hosts: *
 User:ttest_user has Allow permission for operations: Read from hosts: *
```

   If ACL is set for the topic, user **test** does not have the permission to perform the Consumer operation.

   For details about the solution, see **Step 2**.
10. Log in to Kafka Broker using SSH.

    Run the **cd /var/log/Bigdata/kafka/broker** command to go to the log directory.

    Check the **kafka-authorizer.log** file. It shows that the user does not belong to the **kafka** or **kafkaadmin** group.

```
2017-01-25 13:26:33,648 | INFO  | [kafka-request-handler-0] | The principal is test, belongs to Group : [hadoop, ficommon] | kafka.authorizer.logger (SimpleAclAuthorizer.scala:169)
2017-01-25 13:26:33,648 | WARN  | [kafka-request-handler-0] | The user is not belongs to kafka or kafkaadmin group, authorize failed! | kafka.authorizer.logger (SimpleAclAuthorizer.scala:170)
```

For details about the solution, see **Step 3**.

Solution

Step 1 Set **allow.everyone.if.no.acl.found** to **true** and restart the Kafka service.

Step 2 Use the account with permission for login.

Example:

kinit test_user

Alternatively, grant the user with related permission.

NOTICE

This operation must be performed by the Kafka administrator (belonging to the **kafkaadmin** group).

Example:

kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/kafka --topic topic_acl --consumer --add --allow-principal User:test --group test

```
[root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=8.5.144.2:2181/kafka --list --topic topic_acl
Current ACLs for resource `Topic:topic_acl`:
 User:test_user has Allow permission for operations: Describe from hosts: *
 User:test_user has Allow permission for operations: Write from hosts: *
  User:test has Allow permission for operations: Describe from hosts: *
  User:test has Allow permission for operations: Write from hosts: *
  User:test has Allow permission for operations: Read from hosts: *
```

Step 3 Add the user to the **kafka** or **kafkaadmin** group.

----End

16.13.8 SparkStreaming Fails to Consume Kafka Messages, and "Error getting partition metadata" Is Displayed

Symptom

When SparkStreaming is used to consume messages of a specified topic in Kafka, data cannot be obtained from Kafka. The message "Error getting partition metadata" is displayed.

```
Exception in thread "main" org.apache.spark.SparkException:  Error getting partition metadata for 'testtopic'. Does the topic exist?
org.apache.spark.streaming.kafka.KafkaCluster$$anonfun$checkErrors$1.apply(KafkaCluster.scala:366)
org.apache.spark.streaming.kafka.KafkaCluster$$anonfun$checkErrors$1.apply(KafkaCluster.scala:366)
scala.util.Either.fold(Either.scala:97)
org.apache.spark.streaming.kafka.KafkaCluster$.checkErrors(KafkaCluster.scala:365)
org.apache.spark.streaming.kafka.KafkaUtils$.createDirectStream(KafkaUtils.scala:422)
com.xxxxxx.bigdata.spark.examples.FemaleInfoCollectionPrint$.main(FemaleInfoCollectionPrint.scala:45)
com.xxxxxx.bigdata.spark.examples.FemaleInfoCollectionPrint.main(FemaleInfoCollectionPrint.scala)
sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
```

```
java.lang.reflect.Method.invoke(Method.java:498)
org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$
$runMain(SparkSubmit.scala:762)
org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:183)
org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:208)
org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:123)
org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
```

Possible Causes

1. The Kafka service is abnormal.
2. The Consumer client adopts non-security access and access is disabled on the server.
3. The Consumer client adopts non-security access and ACL is set for Kafka topics.

Cause Analysis

1. Check the Kafka service status:
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**.

     Check the Kafka status. It is found that the status is good and the monitoring metrics are correctly displayed.
2. On Manager, check the current Kafka cluster configuration. It is found that **allow.everyone.if.no.acl.found** is not configured or is set to **false**.
   - MRS Manager portal: Log in to MRS Manager and choose **Services** > **Kafka** > **Configuration**.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka** > **Configuration**.
3. If it is set to **false**, the Kafka non-secure port 21005 cannot be used for access.
4. Run the client command to check the ACL permission of the topic.

   ```
   [root@10-10-144-2 client]# kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/
   kafka --list --topic topic_acl
   Current ACLs for resource `Topic:topic_acl`:
    User:test_user has Allow permission for operations: Describe from hosts: *
   User:test_user has Allow permission for operations: Write from hosts: *
   ```

   If an ACL is configured for a topic, the Kafka non-secure port 21005 cannot be used to access the topic.

Solution

Step 1 Add the customized configuration **allow.everyone.if.no.acl.found** or change its value to **true** and restart the Kafka service.

Step 2 Delete the ACL configured for the topic.

Example:

kinit test_user

NOTICE

This operation must be performed by the Kafka administrator (belonging to the **kafkaadmin** group).

Example:

kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/kafka --remove --allow-principal User:test_user --producer --topic topic_acl

kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/kafka --remove --allow-principal User:test_user --consumer --topic topic_acl --group test

----End

16.13.9 Consumer Fails to Consume Data in a Newly Created Cluster, and the Message " GROUP_COORDINATOR_NOT_AVAILABLE" Is Displayed

Symptom

A Kafka cluster is created, and two Broker nodes are deployed. The Kafka client can be used for production but cannot be used for consumption. The Consumer fails to consume data, and the message "GROUP_COORDINATOR_NOT_AVAILABLE" is displayed. The key log is as follows:

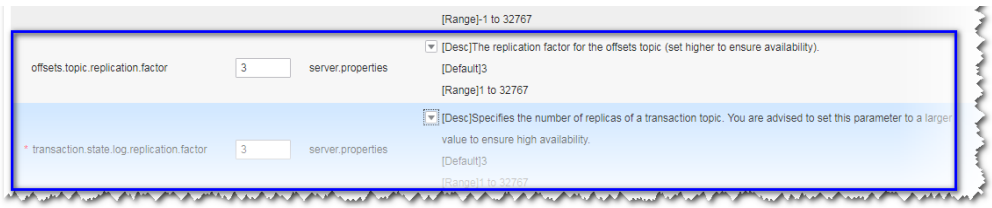
```
2018-05-12 10:58:42,561 | INFO | [kafka-request-handler-3] | [GroupCoordinator 2]: Preparing to restabilize group DemoConsumer with old generation 118 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:59:13,562 | INFO | [executor-Heartbeat] | [GroupCoordinator 2]: Preparing to restabilize group DemoConsumer with old generation 119 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
```

Possible Causes

The **__consumer_offsets** cannot be created.

Cause Analysis

1. As indicated by the log, a large number of **_consumer_offset** creation operations failed.
2. The number of Brokers for the cluster is 2.
3. However, the number of replicas for the **_consumer_offset** topic is 3. Therefore, the creation fails.



Solution

Expand the cluster to at least three streaming core nodes or perform the following steps to modify service configuration parameters:

Step 1 Go to the service configuration page.

- MRS Manager: Log in to MRS Manager, choose **Services** > **Kafka** > **Service Configuration**, and select **All** from **Type**.
- FusionInsight Manager: Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Kafka**. Click **Configurations** and select **All Configurations**.

Step 2 Search for **offsets.topic.replication.factor** and **transaction.state.log.replication.factor** and change their values to **2**.

Step 3 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the services.

----End

16.13.10 SparkStreaming Fails to Consume Kafka Messages, and the Message "Couldn't find leader offsets" Is Displayed

Symptom

When SparkStreaming is used to consume messages of a specified topic in Kafka, data cannot be obtained from Kafka. The following error message is displayed: Couldn't find leader offsets.

```
2018-05-30 12:01:17,816 | INFO  | [Driver] | Reconnect due to socket error: java.net.SocketTimeoutException | kafka.utils.Logging$class.info(Logging.scala:68)
2018-05-30 12:01:47,859 | ERROR | [Driver] | User class threw exception: org.apache.spark.SparkException: java.net.SocketTimeoutException
org.apache.spark.SparkException: Couldn't find leader offsets for Set([STK6,57], [STK6,21]) | org.apache.spark.Logging$class.logError(Logging.scala:96)
org.apache.spark.SparkException: java.net.SocketTimeoutException
org.apache.spark.SparkException: Couldn't find leader offsets for Set([STK6,57], [STK6,21])
        at org.apache.spark.streaming.kafka.KafkaCluster$$anonfun$checkErrors$1.apply(KafkaCluster.scala:366)
        at org.apache.spark.streaming.kafka.KafkaCluster$$anonfun$checkErrors$1.apply(KafkaCluster.scala:366)
        at scala.util.Either.fold(Either.scala:97)
        at org.apache.spark.streaming.kafka.KafkaCluster$.checkErrors(KafkaCluster.scala:365)
        at org.apache.spark.streaming.kafka.KafkaUtils$.createDirectStream(KafkaUtils.scala:422)
        at org.apache.spark.streaming.kafka.KafkaUtils$.createDirectStream(KafkaUtils.scala:532)
        at org.apache.spark.streaming.kafka.KafkaUtils.createDirectStream(KafkaUtils.scala)
        at com.stk.bigdata.sparkstreaming.notify.SparkAlarmControlwz.main(SparkAlarmControlwz.java:194)
        at com.stk.bigdata.sparkstreaming.submit.SparkNotifyA.main(SparkNotifyA.java:14)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.spark.deploy.yarn.ApplicationMaster$$anon$2.run(ApplicationMaster.scala:540)
2018-05-30 12:01:47,863 | INFO  | [Driver] | Final app status: FAILED, exitCode: 15, (reason: User class threw exception: org.apache.spark.SparkException: java.n
org.apache.spark.SparkException: Couldn't find leader offsets for Set([STK6,57], [STK6,21])) | org.apache.spark.Logging$class.logInfo(Logging.scala:59)
2018-05-30 12:01:47,866 | INFO  | [pool-1-thread-1] | Invoking stop() from shutdown hook | org.apache.spark.Logging$class.logInfo(Logging.scala:59)
```

Possible Causes

- The Kafka service is abnormal.
- The network is abnormal.
- The Kafka topic is abnormal.

Cause Analysis

Step 1 On Manager, check the status of the Kafka cluster. The status is **Good**, and the monitoring metrics are correctly displayed.

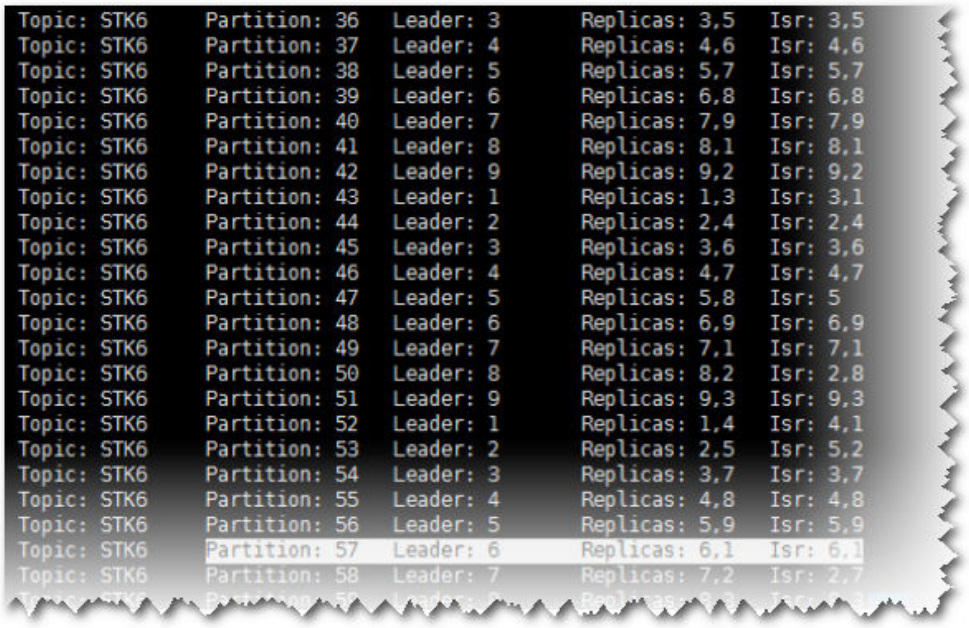
Step 2 View the error topic information in the SparkStreaming log.

Run the Kafka commands to obtain the topic assignment information and copy synchronization information, and check the return result.

kafka-topics.sh --describe --zookeeper <zk_host:port/chroot> --topic <topic name>

If information in the following figure is displayed, the topic is normal. All partitions have normal leader information.

Figure 16-55 Topic distribution information and copy synchronization information



Step 3 Check whether the network connection between the client and Kafka cluster is normal. If no, contact the network team to rectify the fault.

Step 4 Log in to Kafka Broker using SSH.

Run the **cd /var/log/Bigdata/kafka/broker** command to go to the log directory.

Check on **server.log** indicates that the error message is displayed in the log shown in the following figure.

```
2018-05-30 12:02:00,246 | ERROR | [kafka-network-thread-6-PLAINTEXT-3] | Processor got uncaught exception. | kafka.network.Processor (Logging.scala:103)

 java.lang.OutOfMemoryError: Direct buffer memory
 at java.nio.Bits.reserveMemory(Bits.java:694)
 at java.nio.DirectByteBuffer.<init>(DirectByteBuffer.java:123)
 at java.nio.ByteBuffer.allocateDirect(ByteBuffer.java:311)
 at sun.nio.ch.Util.getTemporaryDirectBuffer(Util.java:241)
 at sun.nio.ch.IOUtil.read(IOUtil.java:195)
 at sun.nio.ch.SocketChannelImpl.read(SocketChannelImpl.java:380)
```

at org.apache.kafka.common.network.PlaintextTransportLayer.read(PlaintextTransportLayer.java:110)

Step 5 On Manager, check the configuration of the current Kafka cluster.

- MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka** > **Service Configuration**. Set **Type** to **All**. The value of **-XX:MaxDirectMemorySize** in **KAFKA_JVM_PERFORMANCE_OPTS** is **1G**.
- FusionInsight Manager: Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Kafka** > **Configurations** > **All Configurations**. The value of **-XX:MaxDirectMemorySize** in **KAFKA_JVM_PERFORMANCE_OPTS** is **1G**.

Step 6 If the direct memory is too small, an error is reported. Once the direct memory overflows, the node cannot process new requests. As a result, other nodes or clients fail to access the node due to timeout.

----End

Solution

Step 1 Log in to FusionInsight Manager and go to the Kafka configuration page.

- MRS Manager portal: Log in to MRS Manager and choose **Services** > **Kafka** > **Configuration**.
- FusionInsight Manager: Log in to FusionInsight Manager. Choose Cluster > **Services** > **Kafka** > **Configurations**.

Step 2 Set **Type** to **All**, and search for and change the value of **KAFKA_JVM_PERFORMANCE_OPTS**.

Step 3 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the services.

----End

16.13.11 Consumer Fails to Consume Data and the Message " SchemaException: Error reading field 'brokers'" Is Displayed

Symptom

When a Consumer consumes messages of a specified topic in Kafka, the Consumer cannot obtain data from Kafka. The following error message is displayed: org.apache.kafka.common.protocol.types.SchemaException: Error reading field 'brokers': Error reading field 'host': Error reading string of length 28271, only 593 bytes available.

```
Exception in thread "Thread-0" org.apache.kafka.common.protocol.types.SchemaException: Error reading field 'brokers': Error reading field 'host': Error reading string of length 28271, only 593 bytes available
at org.apache.kafka.common.protocol.types.Schema.read(Schema.java:73)
at org.apache.kafka.clients.NetworkClient.parseResponse(NetworkClient.java:380)
at org.apache.kafka.clients.NetworkClient.handleCompletedReceives(NetworkClient.java:449)
at org.apache.kafka.clients.NetworkClient.poll(NetworkClient.java:269)
at
org.apache.kafka.clients.consumer.internals.ConsumerNetworkClient.clientPoll(ConsumerNetworkClient.java:360)
at
org.apache.kafka.clients.consumer.internals.ConsumerNetworkClient.poll(ConsumerNetworkClient.java:224)
at
org.apache.kafka.clients.consumer.internals.ConsumerNetworkClient.poll(ConsumerNetworkClient.java:192)
at
org.apache.kafka.clients.consumer.internals.ConsumerNetworkClient.poll(ConsumerNetworkClient.java:163)
atorg.apache.kafka.clients.consumer.internals.AbstractCoordinator.ensureCoordinatorReady(AbstractCoordinator.java:179)
at org.apache.kafka.clients.consumer.KafkaConsumer.pollOnce(KafkaConsumer.java:973)
```

```
at org.apache.kafka.clients.consumer.KafkaConsumer.poll(KafkaConsumer.java:937)
at KafkaNew.Consumer$ConsumerThread.run(Consumer.java:40)
```

Possible Causes

The JAR versions of the client and server are inconsistent.

Solution

Modify the Kafka JAR package in the Consumer application to ensure that it is the same as that on the server.

16.13.12 Checking Whether Data Consumed by a Customer Is Lost

Symptom

A Customer saves the consumed data to the database and finds that the data is inconsistent with the production data. Therefore, it is suspected that some of Kafka's consumed data is lost.

Possible Causes

- The customer code is incorrect.
- An exception occurs when Kafka production data is written.
- The Kafka consumption data is abnormal.

Solution

Check Kafka.

Step 1 Observe the changes of the written and consumed offset through **consumer-groups.sh**. (Produce a certain number of messages, and consume these messages on the client to observe the changes of the offset.)

Step 2 Create a consumption group, use the client to consume messages, and view the consumed messages.

new-consumer:

kafka-console-consumer.sh --topic <topic name> --bootstrap-server <IP1:PORT, IP2:PORT,...> --new-consumer --consumer.config <config file>

----End

Check the customer code.

Step 1 Check whether an error is reported when the offset is submitted on the client.

Step 2 If no error is reported, add a printing message to the API that is consumed, and print only the key to view the lost data.

----End

16.13.13 Failed to Start a Component Due to Account Lock

Symptom

In a new cluster, Kafka fails to be started. Authentication failure causes startup failure.

```
/home/omm/kerberos/bin/kinit -k -t /opt/xxxxxx/Bigdata/etc/2_15_ Broker /kafka.keytab kafka/
hadoop.hadoop.com -c /opt/xxxxxx/Bigdata/etc/2_15_ Broker /11846 failed.
export key tab file for kafka/hadoop.hadoop.com failed.export and check keytab file failed, errMsg=]}] for
Broker #192.168.1.92@192-168-1-92.
[2015-07-11 02:34:33] RoleInstance started failure for ROLE[name: Broker].
[2015-07-11 02:34:34] Failed to complete the instances start operation. Current operation entities: [Broker
#192.168.1.92@192-168-1-92], Failure entites : [Broker #192.168.1.92@192-168-1-92].Operation
Failed.Failed to complete the instances start operation. Current operation entities:
[Broker#192.168.1.92@192-168-1-92], Failure entites: [Broker #192.168.1.92@192-168-1-92].
```

Cause Analysis

Check the Kerberos log **/var/log/Bigdata/kerberos/krb5kdc.log**. It is found that IP addresses outside of the cluster uses the **kafka** account for connections, causing multiple authentication failures. As a result, the **kafka** account is locked.

```
Jul 11 02:49:16 192-168-1-91 krb5kdc[1863](info): AS_REQ (2 etypes {18 17}) 192.168.1.93:
NEEDED_PREAUTH: kafka/hadoop.hadoop.com@HADOOP.COM for krbtgt/HADOOP.COM@HADOOP.COM,
Additional pre-authentication required
Jul 11 02:49:16 192-168-1-91 krb5kdc[1863](info): preauth (encrypted_timestamp) verify failure: Decrypt
integrity check failed
Jul 11 02:49:16 192-168-1-91 krb5kdc[1863](info): AS_REQ (2 etypes {18 17}) 192.168.1.93:
PREAUTH_FAILED: kafka/hadoop.hadoop.com@HADOOP.COM for krbtgt/HADOOP.COM@HADOOP.COM,
Decrypt integrity check failed
```

Solution

Log in to a node outside the cluster (for example, 192.168.1.93 in the cause analysis example) and disable Kafka authentication. Wait 5 minutes for the account to be unlocked.

16.13.14 Kafka Broker Reports Abnormal Processes and the Log Shows "IllegalArgumentException"

Symptom

The Process Fault alarm is reported on Manager. Check whether the faulty process is Kafka Broker.

Possible Causes

Broker configuration is abnormal.

Cause Analysis

1. On Manager, obtain the host information on the alarm page.
2. Log in to Kafka Broker using SSH. Run the **cd /var/log/Bigdata/kafka/broker** command to go to the log directory.

   Check the **server.log** file. It is found that the "IllegalArgumentException" exception is thrown in the following log stating "java.lang.IllegalArgumentException: requirement failed: replica.fetch.max.bytes should be equal or greater than message.max.bytes."

   ```
   2017-01-25 09:09:14,930 | FATAL | [main] |  | kafka.Kafka$ (Logging.scala:113)
   java.lang.IllegalArgumentException: requirement failed: replica.fetch.max.bytes should be equal or greater than message.max.bytes
       at scala.Predef$.require(Predef.scala:233)
       at kafka.server.KafkaConfig.validateValues(KafkaConfig.scala:959)
       at kafka.server.KafkaConfig.<init>(KafkaConfig.scala:944)
       at kafka.server.KafkaConfig$.fromProps(KafkaConfig.scala:701)
       at kafka.server.KafkaConfig$.fromProps(KafkaConfig.scala:698)
       at kafka.server.KafkaServerStartable$.fromProps(KafkaServerStartable.scala:28)
       at kafka.Kafka$.main(Kafka.scala:60)
       at kafka.Kafka.main(Kafka.scala)
   ```

   Kafka requires that **replica.fetch.max.bytes** be greater than or equal to **message.max.bytes**.
3. On the Kafka configuration page, select **All Configurations**. All Kafka configurations are displayed. Search for **message.max.bytes** and

   replica.fetch.max.bytes. It is found that the value of **replica.fetch.max.bytes** is less than that of **message.max.bytes**.

Solution

Step 1 Go to the Kafka configuration page.

- For versions earlier than MRS 3.*x*: Log in to MRS Manager and choose **Services** > **Kafka** > **Service Configuration** > **All Configurations**.
- For MRS 3.*x* or later: Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Kafka** > **Configurations** > **All Configurations**.

Step 2 Search for and modify the **replica.fetch.max.bytes** parameter to ensure that its value is greater than or equal to that of **message.max.bytes**. In this way, replicas of partitions on different brokers can be synchronized to all messages.

Step 3 Save the configuration and check whether there is any service whose configuration has expired in the cluster. If yes, restart the corresponding service or role instance for the configuration to take effect.

Step 4 Modify **fetch.message.max.bytes** in the Consumer service application to ensure that the value of **fetch.message.max.bytes** is greater than or equal to that of **message.max.bytes**.

----End

16.13.15 Kafka Topics Cannot Be Deleted

Symptom

When running the following command on the Kafka client to delete topics, it is found that the topics cannot be deleted.

```
kafka-topics.sh --delete --topic test --zookeeper 192.168.234.231:2181/kafka
```

Possible Causes

- The command for connecting the client to ZooKeeper is incorrect.
- Kafka is abnormal and some Kafka nodes are stopped.
- Perform the following operations when Kafka server configurations cannot be deleted.
- Perform the following operations when Kafka configurations are automatically created and the Producer is not stopped.

Cause Analysis

1. After the client command is run, the "ZkTimeoutException" exception is reported.

   ```
   [2016-03-09 10:41:45,773] WARN Can not get the principle name from server 192.168.234.231 (org.apache.zookeeper.ClientCnxn)
   Exception in thread "main" org.I0Itec.zkclient.exception.ZkTimeoutException: Unable to connect to zookeeper server within timeout: 30000
   at org.I0Itec.zkclient.ZkClient.connect(ZkClient.java:880)
   at org.I0Itec.zkclient.ZkClient.<init>(ZkClient.java:98)
   at org.I0Itec.zkclient.ZkClient.<init>(ZkClient.java:84)
   at kafka.admin.TopicCommand$.main(TopicCommand.scala:51)
   at kafka.admin.TopicCommand.main(TopicCommand.scala)
   ```

   For details about the solution, see **Step 1**.

2. Run the following query command on the client:

   kafka-topics.sh --list --zookeeper 192.168.0.122:2181/kafka

   ```
   test - marked for deletion
   ```

   On Manager, check the running status of Kafka Broker instances.

   Run the **cd /var/log/Bigdata/kafka/broker** command to go to the log directory of node **RunningAsController**. Locate **ineligible for deletion: test** in the **controller.log** file.

   ```
   2016-03-09 11:11:26,228 | INFO  | [main] | [Controller 1]: List of topics to be deleted:  | kafka.controller.KafkaController (Logging.scala:68)
   2016-03-09 11:11:26,229 | INFO  | [main] | [Controller 1]: List of topics ineligible for deletion: test | kafka.controller.KafkaController (Logging.scala:68)
   ```

3. On Manager, view the **delete.topic.enable** status of Broker.

   For details about the solution, see **Step 2**.

4. Run the following query command on the client:

   ```
   kafka-topics.sh --describe –topic test --zookeeper 192.168.0.122:2181/kafka
   ```

   Go to the log directory of node **RunningAsController**. Locate **marked ineligible for deletion** in the **controller.log** file.

   ```
   2016-03-10 11:11:17,989 | INFO  | [delete-topics-thread-3] | [delete-topics-thread-3], Handling deletion for topics test | kafka.controller.TopicDeletionManager$DeleteTopicsThread (Logging.scala:68)
   2016-03-10 11:11:17,990 | INFO  | [delete-topics-thread-3] | [delete-topics-thread-3], Not retrying deletion of topic test at this time since it is marked ineligible for deletion | kafka.controller.TopicDeletionManager$DeleteTopicsThread (Logging.scala:68)
   ```

5. On Manager, query the Broker status.

   It can be seen that a Broker is in the Stopped state. In this case, delete the topic and ensure that Brokers where partitions of the topic reside must be in the Good state.

   For details about the solution, see **Step 3**.

6. Go to the log directory of node **RunningAsController**. Locate **Deletion successfully** in the **controller.log** file. If **New topics:[Set(test)]** is displayed again, it indicates that the topic is created again.

```
2016-03-10 11:33:35,208 | INFO  | [delete-topics-thread-3] | [delete-topics-thread-3], Deletion of topic test successfully completed | kafka.controller.TopicDeletionManager$DeleteTopicsThread (Logging.scala:68)

2016-03-10 11:33:38,501 | INFO  | [ZkClient-EventThread-19-192.168.0.122:2181,160.172.0.52:2181,160.172.0.51:2181/kafka] | [TopicChangeListener on Controller 3]: New topics: [Set(test)], deleted topics: [Set()], new partition replica assignment
```

7. Use Manager to query the topic creation configuration of Broker.

   It is confirmed that the application that performs operations on the topic is not stopped.

   For details about the solution, see **Step 4**.

Solution

Step 1 Perform the following operations when connection to ZooKeeper fails.

When the connection between the Kafka client and ZooKeeper times out, run the ping command to check whether the Kafka client can connect to ZooKeeper. Check the network connection between the client and ZooKeeper.

If the network connection fails, check the ZooKeeper service information on Manager.

If ZooKeeper is improperly configured, change the ZooKeeper IP address in the client command.

Step 2 Perform the following operations when Kafka server configurations cannot be deleted.

On Manager, change the value of **delete.topic.enable** to **true**. Save the configurations and restart the service.

The client query command does not contain **Topic:test**.

```
kafka-topics.sh --list --zookeeper 192.168.0.122:24002/kafka
```

Go to the log directory of node **RunningAsController**. Locate **Deletion of topic test successfully** in the **controller.log** file.

```
2016-03-10 10:39:40,665 | INFO  | [delete-topics-thread-3] | [Partition state machine on Controller 3]: Invoking state change to OfflinePartition for partitions [test,2],[test,15],[test,6],[test,16],[test,12],[test,7],[test,10],[test,13],[test,9],[test,19],[test,3],[test,5],[test,1],[test,0],[test,17],[test,8],[test,4],[test,11],[test,14],[test,18] | kafka.controller.PartitionStateMachine (Logging.scala:68)
2016-03-10 10:39:40,668 | INFO  | [delete-topics-thread-3] | [Partition state machine on Controller 3]: Invoking state change to NonExistentPartition for partitions [test,2],[test,15],[test,6],[test,16],[test,12],[test,7],[test,10],[test,13],[test,9],[test,19],[test,3],[test,5],[test,1],[test,0],[test,17],[test,8],[test,4],[test,11],[test,14],[test,18] | kafka.controller.PartitionStateMachine (Logging.scala:68)
2016-03-10 10:39:40,977 | INFO  | [delete-topics-thread-3] | [delete-topics-thread-3], Deletion of topic test successfully completed | kafka.controller.TopicDeletionManager$DeleteTopicsThread (Logging.scala:68)
```

Step 3 Some Kafka nodes are stopped or faulty.

Start the stopped Broker instances.

The client query command does not contain **Topic:test**.

```
kafka-topics.sh --list --zookeeper 192.168.0.122:24002/kafka
```

Go to the log directory of node **RunningAsController**. Locate **Deletion of topic test successfully** in the **controller.log** file.

```
2016-03-10 11:17:56,463 | INFO  | [delete-topics-thread-3] | [Partition state machine on Controller 3]: Invoking state change to NonExistentPartition for partitions [test,4],[test,1],[test,8],[test,2],[test,5],[test,9],[test,7],[test,6],[test,0],[test,3] | kafka.controller.PartitionStateMachine (Logging.scala:68)
2016-03-10 11:17:56,726 | INFO  | [delete-topics-thread-3] | [delete-topics-thread-3], Deletion of topic test successfully completed | kafka.controller.TopicDeletionManager$DeleteTopicsThread (Logging.scala:68)
```

Step 4 Perform the following operations when Kafka configurations are automatically created and the Producer is not stopped.

Stop related applications, change the value of **auto.create.topics.enable** to **false** on Manager. Save the configuration and restart the service.

Step 5 Perform the delete operation again.

----End

16.13.16 Error "AdminOperationException" Is Displayed When a Kafka Topic Is Deleted

Symptom

When running the following command on the Kafka client to set the ACL for a topic, it is found that the ACL cannot be set.

```
kafka-topics.sh --delete --topic test4 --zookeeper 10.5.144.2:2181/kafka
```

The error message "ERROR kafka.admin.AdminOperationException: Error while deleting topic test4" is displayed.

Details are as follows:

```
Error while executing topic command : Error while deleting topic test4
[2017-01-25 14:00:20,750] ERROR kafka.admin.AdminOperationException: Error while deleting topic test4
at kafka.admin.TopicCommand$$anonfun$deleteTopic$1.apply(TopicCommand.scala:177)
at kafka.admin.TopicCommand$$anonfun$deleteTopic$1.apply(TopicCommand.scala:162)
at scala.collection.mutable.ResizableArray$class.foreach(ResizableArray.scala:59)
at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.scala:47)
at kafka.admin.TopicCommand$.deleteTopic(TopicCommand.scala:162)
at kafka.admin.TopicCommand$.main(TopicCommand.scala:68)
at kafka.admin.TopicCommand.main(TopicCommand.scala)
 (kafka.admin.TopicCommand$)
```

Possible Causes

The user does not belong to the **kafkaadmin** group. Kafka provides a secure access interface. Only users in the **kafkaadmin** group can delete topics.

Cause Analysis

1. After the client command is run, the "AdminOperationException" exception is reported.
2. Run the client command **klist** to query the current authenticated user.

   ```
   [root@10-10-144-2 client]# klist
   Ticket cache: FILE:/tmp/krb5cc_0
   Default principal: test@HADOOP.COM

   Valid starting     Expires            Service principal
   01/25/17 11:06:48  01/26/17 11:06:45  krbtgt/HADOOP.COM@HADOOP.COM
   ```

The **test** user is used in this example.

3. Run the **id** command to query the user group information.
   ```
   [root@10-10-144-2 client]# id test
   uid=20032(test) gid=10001(hadoop) groups=10001(hadoop),9998(ficommon),10003(kafka)
   ```

Solution

MRS Manager:

Step 1 Log in to MRS Manager.

Step 2 Choose **System** > **Manage User**.

Step 3 In the **Operation** column of the user, click **Modify**.

Step 4 Add the user to the **kafkaadmin** group. Click **OK**.

Step 5 Run the **id** command to query the user group information.

```
[root@10-10-144-2 client]# id test
uid=20032(test) gid=10001(hadoop)
groups=10001(hadoop),9998(ficommon),10002(kafkaadmin),10003(kafka)
```

----End

FusionInsight Manager:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the target user, and click **Modify**.

Step 4 Add the user to the **kafkaadmin** group. Click **OK**.

Step 5 Run the **id** command to query the user group information.

```
[root@10-10-144-2 client]# id test
uid=20032(test) gid=10001(hadoop)
groups=10001(hadoop),9998(ficommon),10002(kafkaadmin),10003(kafka)
```

----End

16.13.17 When a Kafka Topic Fails to Be Created, "NoAuthException" Is Displayed

Symptom

When running the following command on the Kafka client to create topics, it is found that the topics cannot be created.

```
kafka-topics.sh --create --zookeeper 192.168.234.231:2181/kafka --replication-factor 1 --partitions 2 --topic test
```

Error messages "NoAuthException" and "KeeperErrorCode = NoAuth for /config/topics" are displayed.

Details are as follows:

```
Error while executing topic command org.apache.zookeeper.KeeperException$NoAuthException:
KeeperErrorCode = NoAuth for /config/topics
org.I0Itec.zkclient.exception.ZkException: org.apache.zookeeper.KeeperException$NoAuthException:
KeeperErrorCode = NoAuth for /config/topics
```

```
at org.I0Itec.zkclient.exception.ZkException.create(ZkException.java:68)
at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:685)
at org.I0Itec.zkclient.ZkClient.create(ZkClient.java:304)
at org.I0Itec.zkclient.ZkClient.createPersistent(ZkClient.java:213)
at kafka.utils.ZkUtils$.createParentPath(ZkUtils.scala:215)
at kafka.utils.ZkUtils$.updatePersistentPath(ZkUtils.scala:338)
at kafka.admin.AdminUtils$.writeTopicConfig(AdminUtils.scala:247)
```

Possible Causes

The user does not belong to the **kafkaadmin** group. Kafka provides a secure access interface. Only users in the **kafkaadmin** group can delete topics.

Cause Analysis

1. After the client command is run, the "NoAuthException" exception is reported.

   ```
   Error while executing topic command org.apache.zookeeper.KeeperException$NoAuthException:
   KeeperErrorCode = NoAuth for /config/topics
   org.I0Itec.zkclient.exception.ZkException: org.apache.zookeeper.KeeperException$NoAuthException:
   KeeperErrorCode = NoAuth for /config/topics
    at org.I0Itec.zkclient.exception.ZkException.create(ZkException.java:68)
    at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:685)
    at org.I0Itec.zkclient.ZkClient.create(ZkClient.java:304)
    at org.I0Itec.zkclient.ZkClient.createPersistent(ZkClient.java:213)
    at kafka.utils.ZkUtils$.createParentPath(ZkUtils.scala:215)
    at kafka.utils.ZkUtils$.updatePersistentPath(ZkUtils.scala:338)
    at kafka.admin.AdminUtils$.writeTopicConfig(AdminUtils.scala:247)
   ```

2. Run the client command **klist** to query the current authenticated user.

   ```
   [root@10-10-144-2 client]# klist
   Ticket cache: FILE:/tmp/krb5cc_0
   Default principal: test@HADOOP.COM

   Valid starting     Expires            Service principal
   01/25/17 11:06:48  01/26/17 11:06:45  krbtgt/HADOOP.COM@HADOOP.COM
   ```

   The **test** user is used in this example.

3. Run the **id** command to query the user group information.

   ```
   [root@10-10-144-2 client]# id test
   uid=20032(test) gid=10001(hadoop) groups=10001(hadoop),9998(ficommon),10003(kafka)
   ```

Solution

MRS Manager:

Step 1 Log in to MRS Manager.

Step 2 Choose **System** > **Manage User**.

Step 3 In the **Operation** column of the user, click **Modify**.

Step 4 Add the user to the **kafkaadmin** group.

Step 5 Run the **id** command to query the user group information.

```
[root@10-10-144-2 client]# id test
uid=20032(test) gid=10001(hadoop)
groups=10001(hadoop),9998(ficommon),10002(kafkaadmin),10003(kafka)
```

----End

FusionInsight Manager:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the target user, and click **Modify**.

Step 4 Add the user to the **kafkaadmin** group. Click **OK**.

Step 5 Run the **id** command to query the user group information.

```
[root@10-10-144-2 client]# id test
uid=20032(test) gid=10001(hadoop)
groups=10001(hadoop),9998(ficommon),10002(kafkaadmin),10003(kafka)
```

----End

16.13.18 Failed to Set an ACL for a Kafka Topic, and "NoAuthException" Is Displayed

Symptom

When running the following command on the Kafka client to set the ACL for a topic, it is found that the topic ACL cannot be set.

```
kafka-acls.sh --authorizer-properties zookeeper.connect=10.5.144.2:2181/kafka  --topic topic_acl --producer
--add --allow-principal User:test_acl
```

The error message "NoAuthException: KeeperErrorCode = NoAuth for /kafka-acl-changes/acl_changes_0000000002" is displayed.

Details are as follows:

```
Error while executing ACL command: org.apache.zookeeper.KeeperException$NoAuthException:
KeeperErrorCode = NoAuth for /kafka-acl-changes/acl_changes_0000000002
org.I0Itec.zkclient.exception.ZkException: org.apache.zookeeper.KeeperException$NoAuthException:
KeeperErrorCode = NoAuth for /kafka-acl-changes/acl_changes_0000000002
at org.I0Itec.zkclient.exception.ZkException.create(ZkException.java:68)
at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:995)
at org.I0Itec.zkclient.ZkClient.delete(ZkClient.java:1038)
at kafka.utils.ZkUtils.deletePath(ZkUtils.scala:499)
at kafka.common.ZkNodeChangeNotificationListener$$anonfun$purgeObsoleteNotifications
$1.apply(ZkNodeChangeNotificationListener.scala:118)
at kafka.common.ZkNodeChangeNotificationListener$$anonfun$purgeObsoleteNotifications
$1.apply(ZkNodeChangeNotificationListener.scala:112)
at scala.collection.mutable.ResizableArray$class.foreach(ResizableArray.scala:59)
at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.scala:47)
at
kafka.common.ZkNodeChangeNotificationListener.purgeObsoleteNotifications(ZkNodeChangeNotificationLis
tener.scala:112)
at kafka.common.ZkNodeChangeNotificationListener.kafka$common$ZkNodeChangeNotificationListener$
$processNotifications(ZkNodeChangeNotificationListener.scala:97)
at
kafka.common.ZkNodeChangeNotificationListener.processAllNotifications(ZkNodeChangeNotificationListener
.scala:77)
at kafka.common.ZkNodeChangeNotificationListener.init(ZkNodeChangeNotificationListener.scala:65)
at kafka.security.auth.SimpleAclAuthorizer.configure(SimpleAclAuthorizer.scala:136)
at kafka.admin.AclCommand$.withAuthorizer(AclCommand.scala:73)
at kafka.admin.AclCommand$.addAcl(AclCommand.scala:80)
at kafka.admin.AclCommand$.main(AclCommand.scala:48)
at kafka.admin.AclCommand.main(AclCommand.scala)
Caused by: org.apache.zookeeper.KeeperException$NoAuthException: KeeperErrorCode = NoAuth for /kafka-
acl-changes/acl_changes_0000000002
at org.apache.zookeeper.KeeperException.create(KeeperException.java:117)
at org.apache.zookeeper.KeeperException.create(KeeperException.java:51)
at org.apache.zookeeper.ZooKeeper.delete(ZooKeeper.java:1416)
at org.I0Itec.zkclient.ZkConnection.delete(ZkConnection.java:104)
at org.I0Itec.zkclient.ZkClient$11.call(ZkClient.java:1042)
at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:985)
```

Possible Causes

The user does not belong to the **kafkaadmin** group. Kafka provides a secure access interface. Only users in the **kafkaadmin** group can perform the setting operation.

Cause Analysis

1. After the client command is run, the "NoAuthException" exception is reported.
2. Run the client command **klist** to query the current authenticated user.

   ```
   [root@10-10-144-2 client]# klist
   Ticket cache: FILE:/tmp/krb5cc_0
   Default principal: test@HADOOP.COM

   Valid starting     Expires            Service principal
   01/25/17 11:06:48  01/26/17 11:06:45  krbtgt/HADOOP.COM@HADOOP.COM
   ```

   The **test** user is used in this example.
3. Run the **id** command to query the user group information.

   ```
   [root@10-10-144-2 client]# id test
   uid=20032(test) gid=10001(hadoop) groups=10001(hadoop),9998(ficommon),10003(kafka)
   ```

Solution

MRS Manager:

Step 1 Log in to MRS Manager.

Step 2 Choose **System** > **Manage User**.

Step 3 In the **Operation** column of the user, click **Modify**.

Step 4 Add the user to the **kafkaadmin** group.

Step 5 Run the **id** command to query the user group information.

```
[root@host1 client]# id test
uid=20032(test) gid=10001(hadoop)
groups=10001(hadoop),9998(ficommon),10002(kafkaadmin),10003(kafka)
```

----End

FusionInsight Manager:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **System** > **Permission** > **User**.

Step 3 Locate the row that contains the target user, and click **Modify**.

Step 4 Add the user to the **kafkaadmin** group. Click **OK**.

Step 5 Run the **id** command to query the user group information.

```
[root@10-10-144-2 client]# id test
uid=20032(test) gid=10001(hadoop)
groups=10001(hadoop),9998(ficommon),10002(kafkaadmin),10003(kafka)
```

----End

16.13.19 When a Kafka Topic Fails to Be Created, "NoNode for /brokers/ids" Is Displayed

Symptom

When running the following command on the Kafka client to create topics, it is found that the topics cannot be created.

```
kafka-topics.sh --create  --replication-factor 1 --partitions 2 --topic test --zookeeper 192.168.234.231:2181
```

The error message "NoNodeException: KeeperErrorCode = NoNode for /brokers/ids" is displayed.

Details are as follows:

```
Error while executing topic command : org.apache.zookeeper.KeeperException$NoNodeException: KeeperErrorCode = NoNode for /brokers/ids
[2017-09-17 16:35:28,520] ERROR org.I0Itec.zkclient.exception.ZkNoNodeException: org.apache.zookeeper.KeeperException$NoNodeException: KeeperErrorCode = NoNode for /brokers/ids
    at org.I0Itec.zkclient.exception.ZkException.create(ZkException.java:47)
    at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:995)
    at org.I0Itec.zkclient.ZkClient.getChildren(ZkClient.java:675)
    at org.I0Itec.zkclient.ZkClient.getChildren(ZkClient.java:671)
    at kafka.utils.ZkUtils.getChildren(ZkUtils.scala:541)
    at kafka.utils.ZkUtils.getSortedBrokerList(ZkUtils.scala:176)
    at kafka.admin.AdminUtils$.createTopic(AdminUtils.scala:235)
    at kafka.admin.TopicCommand$.createTopic(TopicCommand.scala:105)
    at kafka.admin.TopicCommand$.main(TopicCommand.scala:60)
    at kafka.admin.TopicCommand.main(TopicCommand.scala)
Caused by: org.apache.zookeeper.KeeperException$NoNodeException: KeeperErrorCode = NoNode for /brokers/ids
    at org.apache.zookeeper.KeeperException.create(KeeperException.java:115)
    at org.apache.zookeeper.KeeperException.create(KeeperException.java:51)
    at org.apache.zookeeper.ZooKeeper.getChildren(ZooKeeper.java:2256)
    at org.apache.zookeeper.ZooKeeper.getChildren(ZooKeeper.java:2284)
    at org.I0Itec.zkclient.ZkConnection.getChildren(ZkConnection.java:114)
    at org.I0Itec.zkclient.ZkClient$4.call(ZkClient.java:678)
    at org.I0Itec.zkclient.ZkClient$4.call(ZkClient.java:675)
    at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:985)
    ... 8 more
 (kafka.admin.TopicCommand$)
```

Possible Causes

- The Kafka service is not running.
- The ZooKeeper address parameter in the client command is incorrectly configured.

Cause Analysis

1. After the client command is run, the "NoNodeException" exception is reported.

   ```
   Error while executing topic command : org.apache.zookeeper.KeeperException$NoNodeException: KeeperErrorCode = NoNode for /brokers/ids
   [2017-09-17 16:35:28,520] ERROR org.I0Itec.zkclient.exception.ZkNoNodeException: org.apache.zookeeper.KeeperException$NoNodeException: KeeperErrorCode = NoNode for /brokers/ids
       at org.I0Itec.zkclient.exception.ZkException.create(ZkException.java:47)
       at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:995)
       at org.I0Itec.zkclient.ZkClient.getChildren(ZkClient.java:675)
       at org.I0Itec.zkclient.ZkClient.getChildren(ZkClient.java:671)
       at kafka.utils.ZkUtils.getChildren(ZkUtils.scala:541)
       at kafka.utils.ZkUtils.getSortedBrokerList(ZkUtils.scala:176)
   ```

```
    at kafka.admin.AdminUtils$.createTopic(AdminUtils.scala:235)
    at kafka.admin.TopicCommand$.createTopic(TopicCommand.scala:105)
    at kafka.admin.TopicCommand$.main(TopicCommand.scala:60)
    at kafka.admin.TopicCommand.main(TopicCommand.scala)
```

2. Check whether the Kafka service is in the normal state on Manager.
3. Check whether the ZooKeeper address in the client command is correct. Check the Kafka information stored in ZooKeeper. The path (Znode) should be suffixed with **/kafka**. It is found that **/kafka** is missing in the configuration.

```
[root@10-10-144-2 client]#
kafka-topics.sh --create  --replication-factor 1 --partitions 2 --topic test --zookeeper
192.168.234.231:2181
```

Solution

Step 1 Ensure that the Kafka service is normal.

Step 2 Add **/kafka** to the ZooKeeper address in the command.

```
[root@10-10-144-2 client]#
kafka-topics.sh --create  --replication-factor 1 --partitions 2 --topic test --zookeeper
192.168.234.231:2181/kafka
```

----End

16.13.20 When a Kafka Topic Fails to Be Created, "replication factor larger than available brokers" Is Displayed

Symptom

When running the following command on the Kafka client to create topics, it is found that the topics cannot be created.

```
kafka-topics.sh --create  --replication-factor 2 --partitions 2 --topic test --zookeeper
192.168.234.231:2181
```

The error message "replication factor larger than available brokers" is displayed.

See the following:

```
Error while executing topic command : replication factor: 2 larger than available brokers: 0
[2017-09-17 16:44:12,396] ERROR kafka.admin.AdminOperationException: replication factor: 2 larger than
available brokers: 0
    at kafka.admin.AdminUtils$.assignReplicasToBrokers(AdminUtils.scala:117)
    at kafka.admin.AdminUtils$.createTopic(AdminUtils.scala:403)
    at kafka.admin.TopicCommand$.createTopic(TopicCommand.scala:110)
    at kafka.admin.TopicCommand$.main(TopicCommand.scala:61)
    at kafka.admin.TopicCommand.main(TopicCommand.scala)
 (kafka.admin.TopicCommand$)
```

Possible Causes

- The Kafka service is not running.
- The available Broker of the Kafka service is smaller than the configured **replication-factor**.
- The ZooKeeper address parameter in the client command is incorrectly configured.

Cause Analysis

1. After the client command is run, "replication factor larger than available brokers" is reported.

```
Error while executing topic command : replication factor: 2 larger than available brokers: 0
[2017-09-17 16:44:12,396] ERROR kafka.admin.AdminOperationException: replication factor: 2 larger than available brokers: 0
    at kafka.admin.AdminUtils$.assignReplicasToBrokers(AdminUtils.scala:117)
    at kafka.admin.AdminUtils$.createTopic(AdminUtils.scala:403)
    at kafka.admin.TopicCommand$.createTopic(TopicCommand.scala:110)
    at kafka.admin.TopicCommand$.main(TopicCommand.scala:61)
    at kafka.admin.TopicCommand.main(TopicCommand.scala)
 (kafka.admin.TopicCommand$)
```

2. Check whether the Kafka service is in the normal state on Manager and whether the current available Broker is smaller than the configured **replication-factor**.
3. Check whether the ZooKeeper address in the client command is correct. Check the Kafka information stored in ZooKeeper. The path (Znode) should be suffixed with **/kafka**. It is found that **/kafka** is missing in the configuration.

```
[root@10-10-144-2 client]#
kafka-topics.sh --create  --replication-factor 2 --partitions 2 --topic test --zookeeper 192.168.234.231:2181
```

Solution

Step 1 Ensure that the Kafka service is in the normal state and the available Broker is not less than the configured **replication-factor**.

Step 2 Add **/kafka** to the ZooKeeper address in the command.

```
[root@10-10-144-2 client]#
kafka-topics.sh --create  --replication-factor 1 --partitions 2 --topic test --zookeeper 192.168.234.231:2181/kafka
```

----End

16.13.21 Consumer Repeatedly Consumes Data

Symptom

When the data volume is large, rebalance occurs frequently, causing repeated consumption. The key logs are as follows:

```
2018-05-12 10:58:42,561 | INFO | [kafka-request-handler-3] | [GroupCoordinator 2]: Preparing to restabilize group DemoConsumer with old generation 118 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:58:43,245 | INFO | [kafka-request-handler-5] | [GroupCoordinator 2]: Stabilized group DemoConsumer generation 119 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:58:43,560 | INFO | [kafka-request-handler-7] | [GroupCoordinator 2]: Assignment received from leader for group DemoConsumer for generation 119 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:59:13,562 | INFO | [executor-Heartbeat] | [GroupCoordinator 2]: Preparing to restabilize group DemoConsumer with old generation 119 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:59:13,790 | INFO | [kafka-request-handler-3] | [GroupCoordinator 2]: Stabilized group DemoConsumer generation 120 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:59:13,791 | INFO | [kafka-request-handler-0] | [GroupCoordinator 2]: Assignment received from leader for group DemoConsumer for generation 120 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 10:59:43,802 | INFO | [kafka-request-handler-2] | Rolled new log segment for '__consumer_offsets-17' in 2 ms. | kafka.log.Log (Logging.scala:68)
2018-05-12 10:59:52,456 | INFO | [group-metadata-manager-0] | [Group Metadata Manager on Broker 2]: Removed 0 expired offsets in 0 milliseconds. | kafka.coordinator.GroupMetadataManager (Logging.scala:68)
2018-05-12 11:00:49,772 | INFO | [kafka-scheduler-6] | Deleting segment 0 from log __consumer_offsets-17.
```

```
| kafka.log.Log (Logging.scala:68)
2018-05-12 11:00:49,773 | INFO | [kafka-scheduler-6] | Deleting index /srv/BigData/kafka/data4/kafka-logs/
__consumer_offsets-17/00000000000000000000.index.deleted | kafka.log.OffsetIndex (Logging.scala:68)
2018-05-12 11:00:49,773 | INFO | [kafka-scheduler-2] | Deleting segment 2147948547 from log
__consumer_offsets-17. | kafka.log.Log (Logging.scala:68)
2018-05-12 11:00:49,773 | INFO | [kafka-scheduler-4] | Deleting segment 4282404355 from log
__consumer_offsets-17. | kafka.log.Log (Logging.scala:68)
2018-05-12 11:00:49,775 | INFO | [kafka-scheduler-2] | Deleting index /srv/BigData/kafka/data4/kafka-logs/
__consumer_offsets-17/00000000002147948547.index.deleted | kafka.log.OffsetIndex (Logging.scala:68)
2018-05-12 11:00:49,775 | INFO | [kafka-scheduler-4] | Deleting index /srv/BigData/kafka/data4/kafka-logs/
__consumer_offsets-17/00000000004282404355.index.deleted | kafka.log.OffsetIndex (Logging.scala:68)
2018-05-12 11:00:50,533 | INFO | [kafka-scheduler-6] | Deleting segment 4283544095 from log
__consumer_offsets-17. | kafka.log.Log (Logging.scala:68)
2018-05-12 11:00:50,569 | INFO | [kafka-scheduler-6] | Deleting index /srv/BigData/kafka/data4/kafka-logs/
__consumer_offsets-17/00000000004283544095.index.deleted | kafka.log.OffsetIndex (Logging.scala:68)
2018-05-12 11:02:21,178 | INFO | [kafka-request-handler-2] | [GroupCoordinator 2]: Preparing to restabilize
group DemoConsumer with old generation 120 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 11:02:22,839 | INFO | [kafka-request-handler-4] | [GroupCoordinator 2]: Stabilized group
DemoConsumer generation 121 | kafka.coordinator.GroupCoordinator (Logging.scala:68)
2018-05-12 11:02:23,169 | INFO | [kafka-request-handler-1] | [GroupCoordinator 2]: Assignment received
from leader for group DemoConsumer for generation 121 | kafka.coordinator.GroupCoordinator
(Logging.scala:68)
2018-05-12 11:02:49,913 | INFO | [kafka-request-handler-6] | Rolled new log segment for
'__consumer_offsets-17' in 2 ms. | kafka.log.Log (Logging.scala:68)
```

In the logs, "Preparing to restabilize group DemoConsumer with old generation" indicates that rebalance occurs.

Possible Causes

The parameter settings are improper.

Cause Analysis

Cause: Due to improper parameter settings, the data processing time is too long when the data volume is large. Balance frequently occurs, and the offset cannot be submitted normally. As a result, the data is repeatedly consumed.

Principle: The offset is submitted only after the poll data is processed. If the processing duration after the poll data is processed exceeds the duration specified by **session.timeout.ms**, the rebalance occurs. As a result, the consumption fails and the offset of the consumed data cannot be submitted. Therefore, the data is consumed at the old offset next time. As a result, the data is repeatedly consumed.

Solution

Adjust the following service parameters on Manager:

request.timeout.ms=100000

session.timeout.ms=90000

max.poll.records=50

heartbeat.interval.ms=3000

Among the preceding parameters:

The value of **request.timeout.ms** is 10s greater than that of **session.timeout.ms**.

The value of **session.timeout.ms** must be within the values of **group.min.session.timeout.ms** and **group.max.session.timeout.ms** on the server.

Set the parameters as required. The **max.poll.records** parameter specifies the number of records for each poll. The purpose is to ensure that the processing time of poll data does not exceed the value of **session.timeout.ms**.

Related Information

- The post-poll data processing must be efficient and do not block the next poll.
- The poll method and data processing suggestion are processed asynchronously.

16.13.22 Leader for the Created Kafka Topic Partition Is Displayed as none

Symptom

When a user creates a topic using the Kafka client command, the leader for the created topic partition is displayed as **none**.

```
[root@10-10-144-2 client]#
kafka-topics.sh --create  --replication-factor 1 --partitions 2 --topic test --zookeeper 10.6.92.36:2181/
kafka

Created topic "test".
```

```
[root@10-10-144-2 client]#
kafka-topics.sh --describe --zookeeper 10.6.92.36:2181/kafka

Topic:test      PartitionCount:2        ReplicationFactor:2     Configs:
        Topic: test     Partition: 0    Leader: none       Replicas: 2,3   Isr:
        Topic: test     Partition: 1    Leader: none       Replicas: 3,1   Isr:
```

Possible Causes

- The Kafka service is not running.
- The user group information cannot be found.

Cause Analysis

1. Check the Kafka service status and monitoring metrics.
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Kafka**. Check the Kafka status. The status is **Good**, and the monitoring metrics are correctly displayed.
2. Obtain the Controller node information on the Kafka overview page.
3. Log in to the node where the Controller resides, and run the **cd /var/log/Bigdata/kafka/broker** command to go to the node log directory. The **state-change.log** contains "NoAuthException", which indicates that the ZooKeeper permission is incorrect.

   ```
   2018-05-31 09:20:42,436 | ERROR | [ZkClient-
   EventThread-34-10.6.92.36:24002,10.6.92.37:24002,10.6.92.38:24002/kafka] | Controller 4 epoch 6
   initiated state change for partition [test,1] from NewPartition to OnlinePartition failed |
   state.change.logger (Logging.scala:103)
   ```

```
 org.I0Itec.zkclient.exception.ZkException: org.apache.zookeeper.KeeperException$NoAuthException:
KeeperErrorCode = NoAuth for /brokers/topics/test/partitions
 at org.I0Itec.zkclient.exception.ZkException.create(ZkException.java:68)
 at org.I0Itec.zkclient.ZkClient.retryUntilConnected(ZkClient.java:1000)
 at org.I0Itec.zkclient.ZkClient.create(ZkClient.java:527)
 at org.I0Itec.zkclient.ZkClient.createPersistent(ZkClient.java:293)
```

4. Check on ZooKeeper audit logs recorded in the specified period also indicates that the permission is abnormal.

```
2018-05-31 09:20:42,421 | ERROR | CommitProcWorkThread-1 | session=0xc3000007015d5a18
user=10.6.92.39,kafka/hadoop.hadoop.com@HADOOP.COM,kafka/
hadoop.hadoop.com@HADOOP.COM    ip=10.6.92.39    operation=create znode
target=ZooKeeperServer    znode=/kafka/brokers/topics/test/partitions/0/state    result=failure
2018-05-31 09:20:42,423 | ERROR | CommitProcWorkThread-1 | session=0xc3000007015d5a18
user=10.6.92.39,kafka/hadoop.hadoop.com@HADOOP.COM,kafka/
hadoop.hadoop.com@HADOOP.COM    ip=10.6.92.39    operation=create znode
target=ZooKeeperServer    znode=/kafka/brokers/topics/test/partitions/0    result=failure
2018-05-31 09:20:42,435 | ERROR | CommitProcWorkThread-1 | session=0xc3000007015d5a18
user=10.6.92.39,kafka/hadoop.hadoop.com@HADOOP.COM,kafka/
hadoop.hadoop.com@HADOOP.COM    ip=10.6.92.39    operation=create znode
target=ZooKeeperServer    znode=/kafka/brokers/topics/test/partitions    result=failure
2018-05-31 09:20:42,439 | ERROR | CommitProcWorkThread-1 | session=0xc3000007015d5a18
user=10.6.92.39,kafka/hadoop.hadoop.com@HADOOP.COM,kafka/
hadoop.hadoop.com@HADOOP.COM    ip=10.6.92.39    operation=create znode
target=ZooKeeperServer    znode=/kafka/brokers/topics/test/partitions/1/state    result=failure
2018-05-31 09:20:42,441 | ERROR | CommitProcWorkThread-1 | session=0xc3000007015d5a18
user=10.6.92.39,kafka/hadoop.hadoop.com@HADOOP.COM,kafka/
hadoop.hadoop.com@HADOOP.COM    ip=10.6.92.39    operation=create znode
target=ZooKeeperServer    znode=/kafka/brokers/topics/test/partitions/1    result=failure
2018-05-31 09:20:42,453 | ERROR | CommitProcWorkThread-1 | session=0xc3000007015d5a18
user=10.6.92.39,kafka/hadoop.hadoop.com@HADOOP.COM,kafka/
hadoop.hadoop.com@HADOOP.COM    ip=10.6.92.39    operation=create znode
target=ZooKeeperServer    znode=/kafka/brokers/topics/test/partitions    result=failure
```

5. Run the **id -Gn kafka** command on each ZooKeeper instance node. It is found that user group information cannot be queried on a node.

```
[root @bdpsit3ap03 ~]# id -Gn kafka
id:  kafka: No such user
[root @bdpsit3ap03 ~]#
```

6. In an MRS cluster, user management is provided by the LDAP service and depends on the SSSD (Red Hat) and NSCD (SUSE) services of OSs. The process from creating a user to synchronizing the user to the SSSD service takes some time. If the user does not take effect or the SSSD version has bugs, the user may be invalid on the ZooKeeper node in some cases, which causes topic creation failures.

Solution

Step 1 Restart the SSD/NSCD service.

- Red Hat

  ```
  service sssd restart
  ```

- SUSE

  ```
  sevice nscd restart
  ```

Step 2 After restarting related services, run the **id username** command on the active ResourceManager node to check whether the user information is valid.

----End

16.13.23 Safety Instructions on Using Kafka

Brief Introduction to API for Kafka

- New Producer API

  Indicates the API defined in org.apache.kafka.clients.producer.KafkaProducer. When kafka-console-producer.sh is used, the API is used by default.

- Old Producer API

  Indicates the API defined in kafka.producer.Producer. When kafka-console-producer.sh is used, the API is invoked to add --old-producer.

- New Consumer API

  Indicates the API defined in org.apache.kafka.clients.consumer.KafkaConsumer. When kafka-console-consumer.sh is used, the API is invoked to add --new-consumer.

- Old Consumer API

  Indicates the API defined in kafka.consumer.ConsumerConnector. When **kafka-console-consumer.sh** is used, the API is used by default.

NOTE

New Producer API and new Consumer API are called new API in general in the document.

Protocol Description for Accessing Kafka

The protocols used to access Kafka are as follows: PLAINTEXT, SSL, SASL_PLAINTEXT, and SASL_SSL.

When Kafka service is started, the listeners using the PLAINTEXT and SASL_PLAINTEXT protocols are started. You can set **ssl.mode.enable** to **true** in Kafka service configuration to start listeners using SSL and SASL_SSL protocols.

The following table describes the four protocols:

Protocol Type	Description	Supported API	Default Port
PLAINTEXT	Supports plaintext access without authentication.	New and old APIs	9092
SASL_PLAINTEXT	Supports plaintext access with Kerberos authentication.	New API	21007
SSL	Supports SSL-encrypted access without authentication.	New API	9093
SASL_SSL	Supports SSL-encrypted access with Kerberos authentication.	New API	21009

ACL Settings for Topic

Kafka supports secure access. Therefore, users can set the ACL for topics to control that different users access different topics. To view and set the permission information about a topic, run the kafka-acls.sh script on the Linux client.

- Scenarios

  Assign Kafka users with specific permissions for related topics based on service requirements.

  The following table describes default Kafka user groups.

User Group	Description
kafkaadmin	Kafka administrator group. Users added to this group have the permissions to create, delete, authorize, as well as read from and write data to all topics.
kafkasuperuser	Users added to this group have permissions to read data from and write data to all topics.
kafka	Kafka common user group. If users in this group want to read data from and write data to a specific topic, the users in the kafkaadmin group must grant permissions to users in this group.

- Prerequisites
  a. The system administrator has understood service requirements and prepared a Kafka administrator (belonging to the kafkaadmin group).
  b. The Kafka client has been installed.
- Procedure
  a. Log in to the node where the Kafka client is installed as the client installation user.
  b. Switch to the Kafka client installation directory, for example, **/opt/kafkaclient**.

     cd /opt/kafkaclient
  c. Run the following command to configure environment variables:

     source bigdata_env
  d. Run the following command to perform user authentication (skip this step for a cluster in common mode):

     kinit *Component service user*
  e. Run the following command to switch to the Kafka client installation directory:

     cd Kafka/kafka/bin
  f. The following describes the commands commonly used for user authorization when **kafka-acl.sh** is used:

- View the permission control list of a topic:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=<ZooKeeper cluster service IP:2181/kafka > --list --topic <Topic name>

- Add the Producer permission for a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=<ZooKeeper cluster service IP:2181/kafka > --add --allow-principal User:<username> --producer --topic <Topic name>

- Remove the Producer permission from a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=<ZooKeeper cluster service IP:2181/kafka > --remove --allow-principal User:<username> --producer --topic <Topic name>

- Add the Consumer permission for a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=<ZooKeeper cluster service IP:2181/kafka > --add --allow-principal User:<username> --consumer --topic <Topic name> --group <consumer group name>

- Remove the Consumer permission from a user:

  ./kafka-acls.sh --authorizer-properties zookeeper.connect=<ZooKeeper cluster service IP:2181/kafka > --remove --allow-principal User:<username> --consumer --topic <Topic name> --group <consumer group name>

Use of New and Old Kafka APIs in Different Scenarios

- Scenario 1: accessing the topic with an ACL

Used API	User Group	Client Parameter	Server Parameter	Access Port
New API	Users need to meet one of the following conditions: • In the administrator group • In the **kafkaadmin** group • In the **kafkasuperuser** group • In the **kafka** group and be authorized	security.protocol=SASL_PLAINTEXT sasl.kerberos.service.name = kafka	-	sasl.port (The default number is 21007.)
		security.protocol=SASL_SSL sasl.kerberos.service.name = kafka	Set ssl.mode.enable to true.	sasl-ssl.port (The default port number is 21009.)
Old API	N/A	N/A	N/A	N/A

- Scenario 2: accessing the topic without an ACL

Used API	User Group	Client Parameter	Server Parameter	Access Port
New API	Users need to meet one of the following conditions: • In the administrator group • In the **kafkaadmin** group • In the **kafkasuperuser** group	security.protocol=SASL_PLAINTEXT sasl.kerberos.service.name = kafka	-	sasl.port (The default number is 21007.)
	Users are in the **kafka** group.		Set **allow.everyone.if.no.acl.found** to **true**.	sasl.port (The default number is 21007.)
	Users need to meet one of the following conditions: • In the administrator group • In the **kafkaadmin** group • In the **kafkasuperuser** group	security.protocol=SASL_SSLsasl.kerberos.service.name = kafka	Set **ssl-enable** to **true**.	sasl-ssl.port (The default port number is 21009.)
	Users are in the **kafka** group.		Set **allow.everyone.if.no.acl.found** to **true**. Set **ssl-enable** to **true**.	sasl-ssl.port (The default port number is 21009.)

Used API	User Group	Client Parameter	Server Parameter	Access Port
	-	security.protocol=PLAINTEXT	Set **allow.everyone.if.no.acl.found** to **true**.	port (The default number is 21005.)
	-	security.protocol=SSL	Set **allow.everyone.if.no.acl.found** to **true**. Set **ssl-enable** to **true**.	ssl.port (The default number is 21008.)
Old Producer	-	-	Set **allow.everyone.if.no.acl.found** to **true**.	port (The default number is 21005.)
Old Consumer	-	-	Set **allow.everyone.if.no.acl.found** to **true**.	ZooKeeper service port: clientPort (The default number is 24002.)

16.13.24 Obtaining Kafka Consumer Offset Information

Symptom

How do I obtain Kafka Consumer offset information when using Kafka Consumer to consume data?

Kafka APIs

- New Producer API

  Indicates the API defined in **org.apache.kafka.clients.producer.KafkaProducer**. When **kafka-console-producer.sh** is used, the API is used by default.

- Old Producer API

  Indicates the API defined in **kafka.producer.Producer**. When **kafka-console-producer.sh** is used, the API is invoked to add **--old-producer**.

- New Consumer API

  Indicates the API defined in **org.apache.kafka.clients.consumer.KafkaConsumer**. When **kafka-console-consumer.sh** is used, the API is invoked to add **--new-consumer**.

- Old Consumer API

Indicates the API defined in **kafka.consumer.ConsumerConnector**. When **kafka-console-consumer.sh** is used, the API is used by default.

NOTE

New Producer API and new Consumer API are called new API in general in the document.

Procedure

Old Consumer API

- Prerequisites
  a. The system administrator has understood service requirements and prepared a Kafka administrator (belonging to the kafkaadmin group).
  b. The Kafka client has been installed.
- Procedure
  a. Log in to the node where the Kafka client is installed as the client installation user.
  b. Switch to the Kafka client installation directory, for example, **/opt/kafkaclient**.

     cd /opt/kafkaclient
  c. Run the following command to configure environment variables:

     source bigdata_env
  d. Run the following command to perform user authentication (skip this step for a cluster in common mode):

     kinit *Component service user*
  e. Run the following command to switch to the Kafka client installation directory:

     cd Kafka/kafka/bin
  f. Run the following command to obtain Consumer offset metric information:

     ```
     bin/kafka-consumer-groups.sh --zookeeper <zookeeper_host:port>/kafka --list
     ```

     ```
     bin/kafka-consumer-groups.sh --zookeeper <zookeeper_host:port>/kafka --describe --group test-consumer-group
     ```

     Example:

     ```
     kafka-consumer-groups.sh --zookeeper 192.168.100.100:2181/kafka --list
     kafka-consumer-groups.sh --zookeeper 192.168.100.100:2181/kafka --describe --group test-consumer-group
     ```

New Consumer API

- Prerequisites
  a. The system administrator has understood service requirements and prepared a Kafka administrator (belonging to the kafkaadmin group).
  b. The Kafka client has been installed.
- Procedure
  a. Log in to the node where the Kafka client is installed as the client installation user.

b. Switch to the Kafka client installation directory, for example, **/opt/client**.

cd /opt/client

c. Run the following command to configure environment variables:

source bigdata_env

d. Run the following command to perform user authentication (skip this step for a cluster in common mode):

kinit *Component service user*

e. Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

f. Run the following command to obtain Consumer offset metric information:

kafka-consumer-groups.sh --bootstrap-server <broker_host:port> --describe --group my-group

Example:

kafka-consumer-groups.sh --bootstrap-server 192.168.100.100:9092 --describe --group my-group

16.13.25 Adding or Deleting Configurations for a Topic

Symptom

Configure or modify a specific topic when using Kafka.

Parameters that can be modified at the topic level:

```
cleanup.policy
compression.type
delete.retention.ms
file.delete.delay.ms
flush.messages
flush.ms
index.interval.bytes
max.message.bytes
min.cleanable.dirty.ratio
min.insync.replicas
preallocate
retention.bytes
retention.ms
segment.bytes
segment.index.bytes
segment.jitter.ms
segment.ms
unclean.leader.election.enable
```

Procedure

- Prerequisites

  The Kafka client has been installed.

- Procedure

  a. Log in to the node where the Kafka client is installed as the client installation user.

  b. Switch to the Kafka client installation directory, for example, **/opt/client**.

cd /opt/client

c. Run the following command to configure environment variables:

source bigdata_env

d. Run the following command to perform user authentication (skip this step for a cluster in common mode):

kinit *Component service user*

e. Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

f. Run the following commands to configure and delete a topic:

kafka-topics.sh --alter --topic <topic_name> --zookeeper <zookeeper_host:port>/kafka --config <name=value>

kafka-topics.sh --alter --topic <topic_name> --zookeeper <zookeeper_host:port>/kafka --delete-config <name>

Example:

kafka-topics.sh --alter --topic test1 --zookeeper 192.168.100.100:2181/kafka --config retention.ms=86400000

kafka-topics.sh --alter --topic test1 --zookeeper 192.168.100.100:2181/kafka --delete-config retention.ms

g. Run the following command to query topic information:

kafka-topics.sh --describe -topic <topic_name> --zookeeper <zookeeper_host:port>/kafka

16.13.26 Reading the Content of the __consumer_offsets Internal Topic

Issue

How does Kafka save the offset of a Consumer to the **__consumer_offsets** of internal topics?

Procedure

Step 1 Log in to the node where the Kafka client is installed as the client installation user.

Step 2 Switch to the Kafka client installation directory, for example, **/opt/client**.

cd /opt/client

Step 3 Run the following command to configure environment variables:

source bigdata_env

Step 4 Run the following command to perform user authentication (skip this step for a cluster in common mode):

kinit *Component service user*

Step 5 Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

Step 6 Run the following command to obtain Consumer offset metric information:

kafka-console-consumer.sh --topic __consumer_offsets --zookeeper <zk_host:port>/kafka --formatter "kafka.coordinator.group.GroupMetadataManager\$OffsetsMessageFormatter" --consumer.config <property file> --from-beginning

Add the following content to the <*property file*> configuration file:

```
exclude.internal.topics = false
```

Example:

kafka-console-consumer.sh --topic __consumer_offsets --zookeeper 10.5.144.2:2181/kafka --formatter "kafka.coordinator.group.GroupMetadataManager\$OffsetsMessageFormatter" --consumer.config ../config/consumer.properties --from-beginning

```
[example-group1,test2,0]::[OffsetMetadata[0,NO_METADATA],CommitTime 1487121209218,ExpirationTime 148720760
9218]
[example-group1,test2,1]::[OffsetMetadata[0,NO_METADATA],CommitTime 1487121209218,ExpirationTime 148720760
9218]
[example-group1,test2,0]::[OffsetMetadata[2,NO_METADATA],CommitTime 1487121269208,ExpirationTime 148720760
9208]
[example-group1,test2,1]::[OffsetMetadata[1,NO_METADATA],CommitTime 1487121269208,ExpirationTime 148720760
```

----End

16.13.27 Configuring Logs for Shell Commands on the Client

Issue

How do I set the log level for shell commands on the client?

Procedure

Step 1 Log in to the node where the Kafka client is installed as the client installation user.

Step 2 Switch to the Kafka client installation directory, for example, **/opt/client**.

cd /opt/client

Step 3 Run the following command to switch to the Kafka client configuration directory:

cd Kafka/kafka/config

Step 4 Open the **tools-log4j.properties** file, change **WARN** to **INFO**, and save the file.

```
log4j.rootLogger=WARN, stderr

log4j.appender.stderr=org.apache.log4j.ConsoleAppender
log4j.appender.stderr.layout=org.apache.log4j.PatternLayout
log4j.appender.stderr.layout.ConversionPattern=[%d] %p %m (%c)%n
log4j.appender.stderr.Target=System.err
```

```
log4j.rootLogger=INFO, stderr

log4j.appender.stderr=org.apache.log4j.ConsoleAppender
log4j.appender.stderr.layout=org.apache.log4j.PatternLayout
log4j.appender.stderr.layout.ConversionPattern=[%d] %p %m (%c)%n
log4j.appender.stderr.Target=System.err
```

Step 5 Switch to the Kafka client installation directory, for example, **/opt/client**.

cd /opt/client

Step 6 Run the following command to configure environment variables:

source bigdata_env

Step 7 Run the following command to perform user authentication (skip this step for a cluster in common mode):

kinit *Component service user*

Step 8 Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

Step 9 Run the following command to obtain the topic information. The log information can be viewed on the console.

```
kafka-topics.sh --list --zookeeper 10.5.144.2:2181/kafka
[2017-02-17 14:34:27,005] INFO JAAS File name: /opt/client/Kafka/./kafka/config/jaas.conf
(org.I0Itec.zkclient.ZkClient)
[2017-02-17 14:34:27,007] INFO Starting ZkClient event thread. (org.I0Itec.zkclient.ZkEventThread)
[2017-02-17 14:34:27,013] INFO Client environment:zookeeper.version=V100R002C10, built on 05/12/2016
08:56 GMT (org.apache.zookeeper.ZooKeeper)
[2017-02-17 14:34:27,013] INFO Client environment:host.name=10-10-144-2
(org.apache.zookeeper.ZooKeeper)
[2017-02-17 14:34:27,013] INFO Client environment:java.version=1.8.0_72
(org.apache.zookeeper.ZooKeeper)
[2017-02-17 14:34:27,013] INFO Client environment:java.vendor=Oracle Corporation
(org.apache.zookeeper.ZooKeeper)
[2017-02-17 14:34:27,013] INFO Client environment:java.home=/opt/client/JDK/jdk/jre
(org.apache.zookeeper.ZooKeeper)
Test
__consumer_offsets
counter
test
test2
test3
test4
```

----End

16.13.28 Obtaining Topic Distribution Information

Issue

How do I obtain topic distribution information in a Broker instance?

Preparations

- Prerequisites

  The Kafka and ZooKeeper clients have been installed.

- Procedure

  a. Log in to the node where the Kafka client is installed as the client installation user.

  b. Switch to the Kafka client installation directory, for example, **/opt/client**.

  cd /opt/client

  c. Run the following command to configure environment variables:

source bigdata_env

d. Run the following command to perform user authentication (skip this step for a cluster in common mode):

kinit *Component service user*

e. Run the following command to switch to the Kafka client installation directory:

cd Kafka/kafka/bin

f. Run the Kafka commands to obtain the topic assignment information and copy synchronization information, and check the return result.

kafka-topics.sh --describe --zookeeper <zk_host:port/chroot>

Example:

```
[root@mgtdat-sh-3-01-3 client]#kafka-topics.sh --describe --zookeeper 10.149.0.90:2181/
kafka
Topic:topic1   PartitionCount:2  ReplicationFactor:2     Configs:
Topic: topic1  Partition: 0 Leader: 26 Replicas: 23,25 Isr: 26
Topic: topic1  Partition: 1 Leader: 24 Replicas: 24,23 Isr: 24,23
```

In the preceding information, **Replicas** indicates the replica assignment information and **Isr** indicates the replica synchronization information.

Solution 1

1. Query the Broker ID mapping in ZooKeeper.

   sh zkCli.sh -server <zk_host:port>

2. Run the following command on the ZooKeeper client:

   ls /kafka/brokers/ids

   get/kafka/brokers/ids/<queried Broker ID>

   Example:

```
[root@node-master1gAMQ kafka]# zkCli.sh -server node-master1gAMQ:2181
Connecting to node-master1gAMQ:2181
Welcome to ZooKeeper!
JLine support is enabled

WATCHER::

WatchedEvent state:SyncConnected type:None path:null
[zk: node-master1gAMQ:2181(CONNECTED) 0] ls /kafka/brokers/
ids     seqid    topics
[zk: node-master1gAMQ:2181(CONNECTED) 0] ls /kafka/brokers/ids
[1]
[zk: node-master1gAMQ:2181(CONNECTED) 1] get /kafka/brokers/ids/1
{"listener_security_protocol_map":{"PLAINTEXT":"PLAINTEXT","SSL":"SSL"},"endpoints":["PLAINTEXT://
192.168.2.242:9092","SSL://192.168.2.242:9093"],"rack":"/default/
rack0","jmx_port":21006,"host":"192.168.2.242","timestamp":"1580886124398","port":9092,"version":4}
[zk: node-master1gAMQ:2181(CONNECTED) 2]
```

Solution 2

Obtain the mapping between nodes and Broker IDs.

kafka-broker-info.sh --zookeeper <zk_host:port/chroot>

Example:

```
[root@node-master1gAMQ kafka]# bin/kafka-broker-info.sh --zookeeper 192.168.2.70:2181/kafka
Broker_ID     IP_Address
```

```
---------------------------
1          192.168.2.242
```

16.13.29 Kafka HA Usage Description

Kafka High Reliability and Availability

Kafka message transmission assurance mechanism ensures message transmission after required parameters are set to meet different performance and reliability requirements.

- **Kafka high availability and high performance**

  If HA and high performance are required, configure parameters listed in the following table.

Parameter	Default Value	Description
unclean.leader.election.enable	true	Specifies whether a replica that is not in the ISR can be selected as the leader. If this parameter is set to **true**, data may be lost.
auto.leader.rebalance.enable	true	Specifies whether the leader automated balancing function is used. If this parameter is set to **true**, the controller periodically balances the leader of each partition on all nodes and assigns the leader to a replica with a higher priority.

Parameter	Default Value	Description
acks	1	The leader needs to check whether the message has been received and determine whether the required operation has been processed. This parameter affects message reliability and performance. • If this parameter is set to **0**, the Producer does not wait for any response from the server and the message is considered successful. • If this parameter is set to **1**, when the leader of the copy verifies that data has been written into the cluster, the leader makes repose quickly without waiting until all the copies are written. In this case, if the leader is abnormal when the leader makes the confirmation but replica synchronization is not complete, data will be lost. • If this parameter is set to **-1** (all), the synchronization is successful only after all synchronization copies are confirmed. If **min.insync.replicas** is also configured, multiple copies can be written successfully. In this case, as long as one copy remains active, the record is not lost. **NOTE** This parameter is configured in the Kafka client configuration file.
min.insync.replicas	1	Specifies the minimum number of replicas to which data is written when **acks** is set to **-1** for the Producer.

Impact of HA and high performance configurations:

NOTICE

After HA and high performance are configured, the data reliability decreases. Specifically, data may be lost of disks or nodes are faulty.

- **Kafka high reliability configuration**

  If high data reliability is required, configure parameters listed in the following table.

Parameter	Recommended Value	Description
unclean.leader.election.enable	false	Indicates whether a replica that is not in the ISR list can be elected as a leader.
acks	-1	The leader needs to check whether the message has been received and determine whether the required operation has been processed. If this parameter is set to **-1**, the message is successfully received only when all replicas in the ISR list have confirmed to receive the message. The **min.insync.replicas** parameter must also be set to ensure that multiple copies can be written successfully. As long as one copy is active, the record is not lost. **NOTE** This parameter is configured in the Kafka client configuration file.
min.insync.replicas	2	Specifies the minimum number of replicas to which data is written when **acks** is set to **-1** for the Producer. Ensure that the value of **Min.insync.replicas** is equal to or less than that of **replication.factor**.

Impact of high reliability configurations:

- Deteriorated performance

  All copies in the ISR list are required, and the writing of the minimum number of copies has been verified successful. As a result, the delay of a single message increases and the processing capability of the client decreases. The actual performance depends on the onsite test data.

- Reduced availability

  A replica that is not in the ISR list cannot be elected as a leader. If the leader goes offline and other replicas are not in the ISR list, the partition remains unavailable until the leader node recovers.

  All copies in the ISR list are required, and the writing of the minimum number of copies has been verified successful. When the node where a copy of a partition is located is faulty, the minimum number of successful copies cannot be met. As a result, service writing fails.

Configuration Impact

Evaluate reliability and performance requirements based on service scenarios and use proper parameter configuration.

NOTE

- For valuable data, you are advised to configure raid1 or raid5 for Kafka data directory disks to improve data reliability in case disk fault of a single disk.
- The **acks** parameter is named different for different Producer APIs.
  - New Producer API

    Indicates the interface defined in **org.apache.kafka.clients.producer.KafkaProducer**. The **acks** parameter name remains unchanged for this API.
  - Old Producer API

    Indicates the interface defined in **kafka.producer.Producer**. The **acks** parameter is named as **request.required.acks** for this API.
- For parameters that can be modified at the topic level, the service level configurations are used by default. These parameters can be separately configured based on topic reliability requirements.

  For example, you can configure the reliability parameters of the topic named **test**.

  kafka-topics.sh --zookeeper 192.168.1.205:2181/kafka --alter --topic test --config unclean.leader.election.enable=false --config min.insync.replicas=2

  192.168.1.205 indicates the ZooKeeper service IP address.
- If modification of the service-level requires the restart of Kafka, you are advised to modify the service-level configuration on the change page.

16.13.30 Kafka Producer Writes Oversized Records

Symptom

When a user develops a Kafka application and invokes the new interface (**org.apache.kafka.clients.producer.***) as a Producer to write data to Kafka, the size of a single record is 1100055, which exceeds the value (**1000012**) of **message.max.bytes** in the Kafka configuration file **server.properties**. After the values of **message.max.bytes** and **replica.fetch.max.bytes** in the Kafka service configuration are changed to **5242880**, the exception persists. The error information is as follows:

```
.....
14749 [Thread-0] INFO  com.xxxxxx.bigdata.kafka.example.NewProducer  - The ExecutionException occured : {}.
java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.RecordTooLargeException: The message is 1100093 bytes when serialized which is larger than the maximum request size you have configured with the max.request.size configuration.
at org.apache.kafka.clients.producer.KafkaProducer$FutureFailure.<init>(KafkaProducer.java:739)
at org.apache.kafka.clients.producer.KafkaProducer.doSend(KafkaProducer.java:483)
at org.apache.kafka.clients.producer.KafkaProducer.send(KafkaProducer.java:430)
at org.apache.kafka.clients.producer.KafkaProducer.send(KafkaProducer.java:353)
at com.xxxxxx.bigdata.kafka.example.NewProducer.run(NewProducer.java:150)
Caused by: org.apache.kafka.common.errors.RecordTooLargeException: The message is **** bytes when serialized which is larger than the maximum request size you have configured with the max.request.size configuration.
.....
```

Cause Analysis

When data is written to Kafka, the Kafka client compares the value of **max.request.size** with the size of the data to be written. If the size of the data to be written exceeds the default value of **max.request.size**, the preceding exception is reported.

Solution

Step 1 You can set the value of **max.request.size** when initializing the Kafka Producer instance.

For example, you can set this parameter to **5252880** as follows:

```
// Protocol type: Currently, the SASL_PLAINTEXT or PLAINTEXT protocol types can be used.
props.put(securityProtocol, kafkaProc.getValues(securityProtocol, "SASL_PLAINTEXT"));
 // Service name
props.put(saslKerberosServiceName, "kafka");
props.put("max.request.size", "5252880");
.....
```

----End

16.13.31 Kafka Consumer Reads Oversized Records

Symptom

After data is written to Kafka, a user develops an application and invokes the interface (**org.apache.kafka.clients.consumer.***) to read data from Kafka as a Consumer. However, the reading fails and the following error is reported:

```
.....
1687 [KafkaConsumerExample] INFO  org.apache.kafka.clients.consumer.internals.AbstractCoordinator  - Successfully joined group DemoConsumer with generation 1
1688 [KafkaConsumerExample] INFO  org.apache.kafka.clients.consumer.internals.ConsumerCoordinator  - Setting newly assigned partitions [default-0, default-1, default-2] for group DemoConsumer
2053 [KafkaConsumerExample] ERROR com.xxxxxx.bigdata.kafka.example.NewConsumer  - [KafkaConsumerExample], Error due to
org.apache.kafka.common.errors.RecordTooLargeException: There are some messages at [Partition=Offset]: {default-0=177} whose size is larger than the fetch size 1048576 and hence cannot be ever returned. Increase the fetch size on the client (using max.partition.fetch.bytes), or decrease the maximum message size the broker will allow (using message.max.bytes).
2059 [KafkaConsumerExample] INFO  com.xxxxxx.bigdata.kafka.example.NewConsumer  - [KafkaConsumerExample], Stopped
.....
```

Cause Analysis

When reading data, the Kafka client compares the size of the data to be read with the value of **max.partition.fetch.bytes**. If the size exceeds the value of **max.partition.fetch.bytes**, the preceding exception is reported.

Solution

Step 1 When creating a Kafka Consumer instance during initialization, set **max.partition.fetch.bytes**.

For example, you can set this parameter to **5252880** as follows:

```
.....
// Security protocol type
props.put(securityProtocol, kafkaProc.getValues(securityProtocol, "SASL_PLAINTEXT"));
// Service name
props.put(saslKerberosServiceName, "kafka");

props.put("max.partition.fetch.bytes","5252880");
.....
```

----End

16.13.32 High Usage of Multiple Disks on a Kafka Cluster Node

Issue

The usage of multiple disks on a node in the Kafka streaming cluster is high. The Kafka service will become unavailable if the usage reaches 100%.

Symptom

A node in the MRS Kafka streaming cluster by the customer has multiple disks. Due to improper partitioning and service reasons, the usage of some disks is high. When the usage reaches 100%, Kafka becomes unavailable.

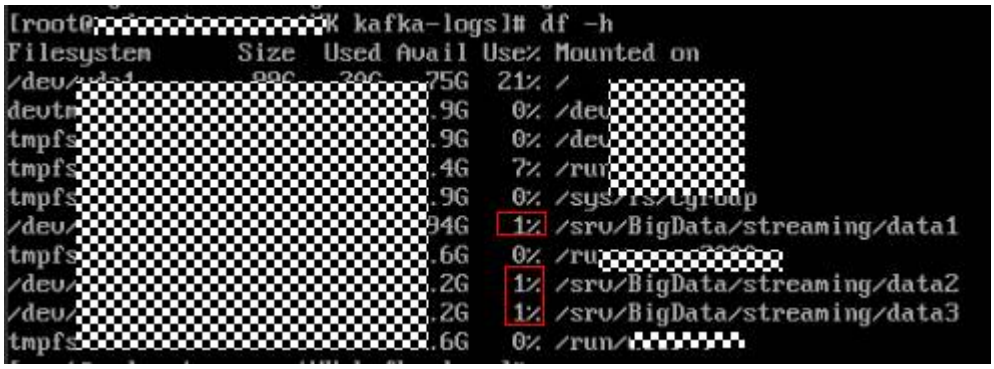
Cause Analysis

The disk data needs to be processed in a timely manner. After the value of **log.retention.hours** is changed, the service needs to be restarted. To ensure service continuity, you can shorten the aging time of a single data-intensive topic as required.

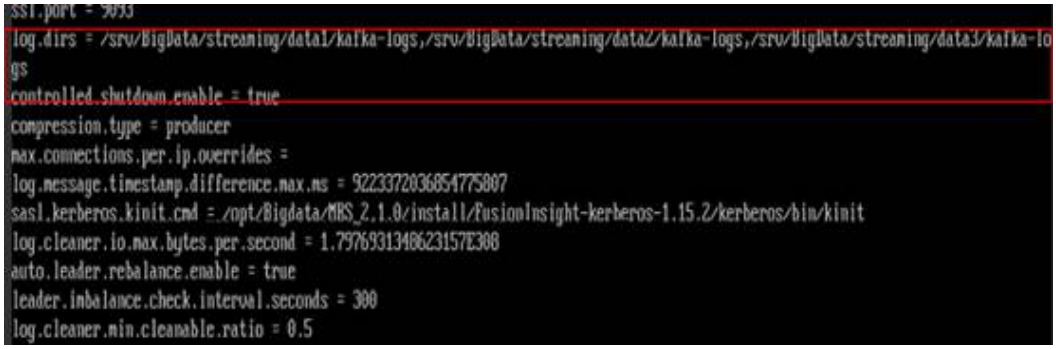
Procedure

Step 1 Log in to the core node of the Kafka streaming cluster.

Step 2 Run the **df –h** command to check the disk usage.

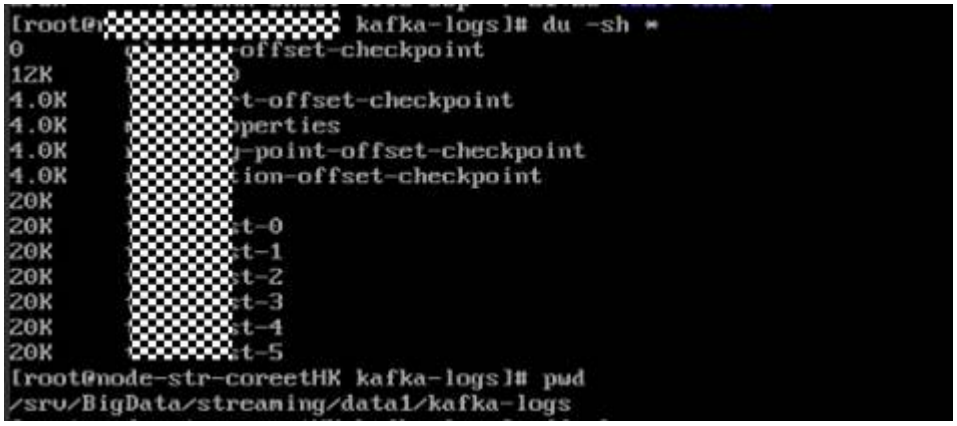


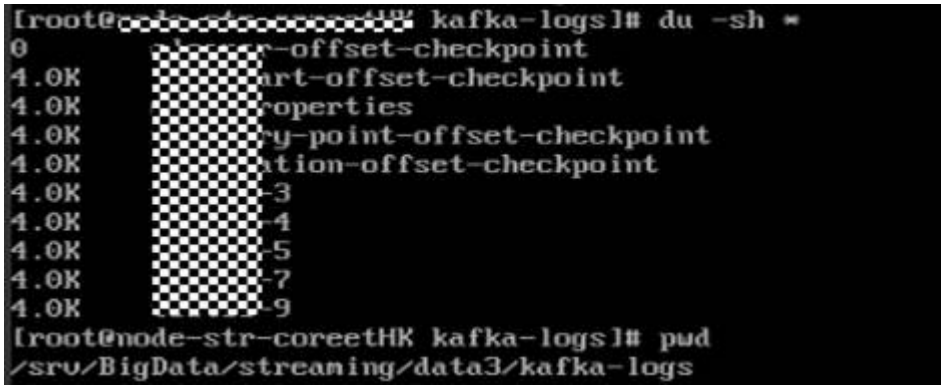
Step 3 Obtain the data storage directory from the **log.dirs** configuration item in the Kafka configuration file **opt/Bigdata/MRS_2.1.0/1_11_Broker/etc/server.properties**. Change the configuration file path based on the cluster version in the environment. If there are multiple disks, use commas (,) to separate multiple configuration items.



Step 4 Run the **cd** command to go to the data storage directory obtained in **Step 3** of the disk with high usage.

Step 5 Run the **du -sh *** command to print the name and size of the current topic.





Step 6 Determine the method of changing the data retention period. The default global data retention period of Kafka is seven days. A large amount of data may be written to some topics, and these topics reside on the partitions on the disk with high usage.

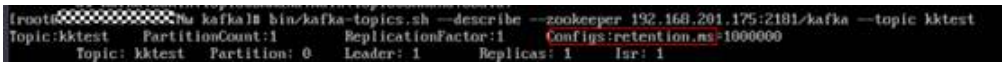
- You can change the global data retention period to a smaller value to release disk space. This method requires a Kafka service restart, which may affect service running. For details, see **Step 7**.
- You can change the data retention period of a single topic to a smaller value to release disk space. This configuration takes effect without a Kafka service restart. For details, see **Step 8**.

Step 7 Log in to Manager. On the Kafka service configuration page, switch to **All Configurations** and search for the **log.retention.hours** configuration item. The default value is 7 days. Change it based on the site requirements.

Step 8 Change the data retention time of the topics on these disks.

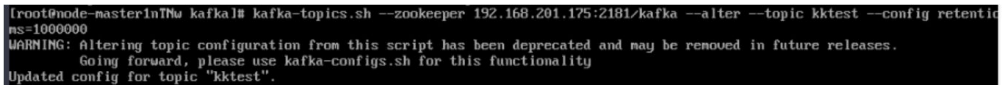
1. Check the retention time of the topic data.

   bin/kafka-topics.sh --describe --zookeeper *<ZooKeeper cluster service IP address>*:2181/kafka --topic kktest

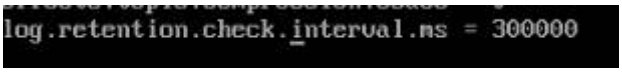
   

2. Set the topic data retention time. **--topic** indicates the topic name, and **retention.ms** indicates the data retention time, in milliseconds.

   kafka-topics.sh --zookeeper *<ZooKeeper cluster service IP address>*:2181/kafka --alter --topic kktest --config retention.ms=1000000

   After the data retention time is set, the deletion operation may not be performed immediately. The deletion operation starts after the time specified by **log.retention.check.interval.ms**. You can check whether the **delete** field exists in the **server.log** file of Kafka to determine whether the deletion operation takes effect. If the **delete** field exists, the deletion operation has taken effect. You can also run the **df –h** command to check the disk usage and determine whether the setting takes effect.

----End

16.14 Using Oozie

16.14.1 Oozie Jobs Do Not Run When a Large Number of Jobs Are Submitted Concurrently

Issue

When a large number of Oozie jobs are submitted concurrently, the jobs do not run.

Symptom

When a large number of Oozie jobs are submitted concurrently, the jobs do not run.

Cause Analysis

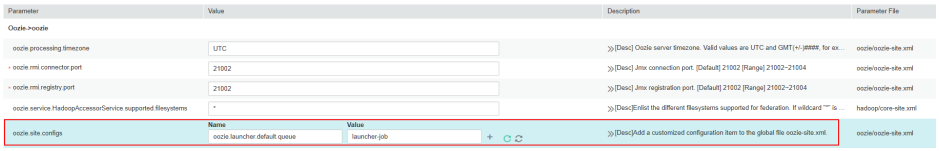
When Oozie submits a job, an oozie-launcher job is started first, and then the oozie-launcher job submits the real job for execution. By default, the oozie-launcher job and the real job are in the same queue.

When a large number of Oozie jobs are submitted concurrently, a large number of oozie-launcher jobs may be started, exhausting the resources of the queue. As a result, no more resources are available to start real jobs, and the jobs are not executed.

Procedure

Step 1 Create a queue for Oozie. For details, see **User Guide** > **Managing an Existing Cluster** > **Tenant Management** > **Creating a Tenant**. You can also use the launcher-job queue generated during MRS cluster creation.

Step 2 On Manager, choose **Cluster** > **Services** > **Oozie** > **Configurations**, search for **oozie.site.configs**, and add **oozie.launcher.default.queue** as the parameter name and **launcher-job** as the value.



----End

16.15 Using Presto

16.15.1 During sql-standard-with-group Configuration, a Schema Fails to Be Created and the Error Message "Access Denied" Is Displayed

Issue

A schema fails to be created during sql-standard-with-group configuration and the error message "Access Denied" is displayed.

Symptom

```
CREATE SCHEMA hive.sf2 WITH (location = 'obs://obs-zy1234/sf2');Query 20200224_031203_00002_g6gzy failed: Access Denied: Cannot create schema sf2
```

Cause Analysis

To create a schema in Presto, you must have the administrator permission of Hive.

Procedure

MRS Manager:

- Method 1:
  a. Log in to MRS Manager and choose **System** > **Manage User**.
  b. Locate the row that contains the target user, and click **Modify** in the **Operation** column.

c. Click **Select and Add Role** to assign the **System_administrator** permission to the user.

d. Click **OK**.

- Method 2:

a. Log in to MRS Manager and choose **System** > **Manage Role**.

b. Click **Create Role** and set the following parameters:

- Enter a role name, for example, **hive_admin**.
- Set **Permission** to **Hive** and select **Hive Admin Privilege**.

c. Click **OK** to save the role.

d. Choose **System** > **Manage User**.

e. Locate the row that contains the target user, and click **Modify** in the **Operation** column.

f. Click **Select and Add Role** to add the newly created hive_admin permission to the user.

g. Click **OK**.

FusionInsight Manager:

- Method 1:

a. Log in to FusionInsight Manager and choose **System** > **Permission** > **User**.

b. Locate the row that contains the target user, and click **Modify** in the **Operation** column.

c. Click **Add** next to the role to assign the **System_administrator** permission to the user.

d. Click **OK**.

- Method 2:

a. Log in to FusionInsight Manager and choose **System** > **Permission** > **Role**.

b. Click **Create Role** and set the following parameters:

- Enter a role name, for example, **hive_admin**.
- To configure resource permissions, select **Hive** and **Hive Admin Permissions**.

c. Click **OK** to save the role.

d. Choose **System** > **Permission** > **User**.

e. Locate the row that contains the target user, and click **Modify** in the **Operation** column.

f. Click **Add** next to the role to add the **hive_admin** permission for the user.

g. Click **OK**.

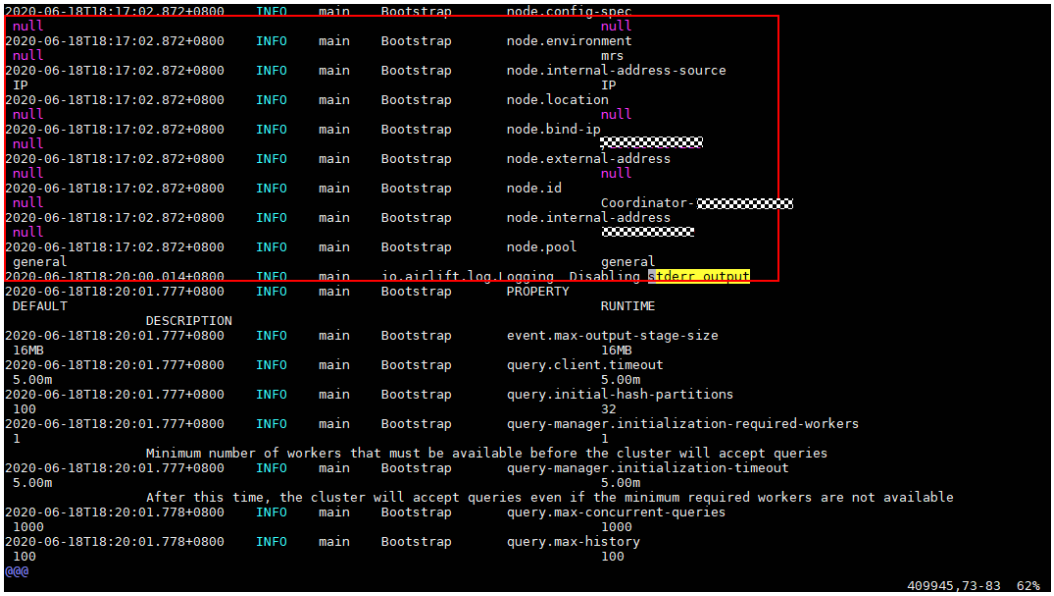
16.15.2 The Presto coordinator cannot be started properly.

Issue

The coordinator process of Presto is killed due to an unknown reason, or the coordinator process of Presto cannot be started.

Symptom

The Presto coordinator process cannot be started properly. On the Manager page, it is shown that the presto coordinator process is started properly and its status is normal. However, the background log shows that the coordinator process is not started. Only the following log is displayed:



The Presto coordinator is killed before being started, and no other logs are printed. Other Presto logs do not indicate the reason why the presto coordinator is killed.

Cause Analysis

The port check logic of the presto health check script does not distinguish ports.

Procedure

Step 1 Use a tool to log in to the master nodes of the cluster and perform the following operations:

Step 2 Run the following command to edit the file:

vim /opt/Bigdata/MRS_xxx/install/FusionInsight-Presto-*/ha/module/harm/plugin/script/pcd.sh

Change line 31 in the file to **http_port_exists=$(netstat -apn | awk '{print $4, $6}' | grep :${HTTP_PORT} | grep LISTEN | wc -l)**.

```
25
26 check_status()
27 {
28         proc_exists=$(ps -ef | grep com.facebook.presto.server.PrestoServer | grep -v grep | wc -l)
29         param="-u $PRESTO_SERVER/v1/cluster"
30         if [[ ${proc_exists} == 1 ]]; then
31             http_port_exists=$(netstat -apn | awk '{print $4, $6}' | grep :${HTTP_PORT} | grep LISTEN | wc -l)
32
33             if [[ ${http_port_exists} == 1 ]] ;then
34                 log ${PCD_LOG_FILE} "INFO" "return [ normal ]"
35                 return 0
36             else
37                 log ${PCD_LOG_FILE} "ERROR" "HTTP PORT does not exist, return [ abnormal ]"
38                 return 2
39             fi
40         else
41             log ${PCD_LOG_FILE} "INFO" " coordinator process not exists, return [ abnormal ]"
42             return 2
43         fi
44 }
45
```

Step 3 Save the modification. On FusionInsight Manager, choose **Services** > **Presto** > **Instances** to restart the Coordinator process.

----End

16.15.3 An Error Is Reported When Presto Is Used to Query a Kudu Table

Issue

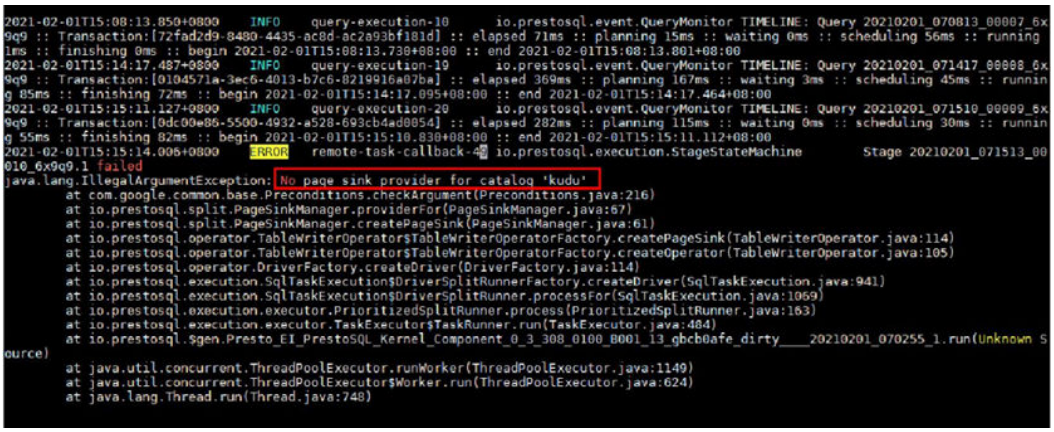
An error is reported when Presto is used to query a Kudu table.

Symptom

When Presto is used to query a Kudu table, the following error message is displayed.

```
presto:default> show tables;
          Table

 impala::default.kudu_taobao
 impala::default.kudu_tt
 impala::default.kudutest
(3 rows)

Query 20210201_030636_00026_95mzd, FINISHED, 4 nodes
Splits: 53 total, 53 done (100.00%)
0:00 [3 rows, 125B] [18 rows/s, 766B/s]

presto:default> select count(*) from kudu.default.kudu_taobao;
Query 20210201_030653_00027_95mzd failed: line 1:22: Table kudu.default.kudu_taobao does not exist
select count(*) from kudu.default.kudu_taobao

presto:default> select count(*) from kudu_taobao;
Query 20210201_030939_00028_95mzd failed: line 1:22: Table kudu.default.kudu_taobao does not exist
select count(*) from kudu_taobao

presto:default>
```

Error information



Cause Analysis

There are no Kudu configurations on the actually running node (node where the worker instance is located).

Procedure

Step 1 Add configuration file **kudu.properties** to all worker instance nodes in the Presto cluster.

Path for storing the configuration file: **/opt/Bigdata/MRS_xxx/1_x_Worker/etc/catalog/** (Change the path based on the actual cluster version.)

Configuration file content:

```
connector.name=kudu
kudu.client.master-addresses=KuduMasterIP1:port,KuduMasterIP2:port,KuduMasterIP3:port
```

NOTE

- Set the IP address and port number of the KuduMaster node based on the site requirements.
- Add the file permission and owner group that are the same as those of other files in the file save path to the configuration file.

Step 2 After the modification, choose **Components** > **Kudu** on the cluster details page, click **More**, and select **Restart Service**.

----End

16.15.4 No Data is Found in the Hive Table Using Presto

Issue

When Presto is used to query the Hive table, no data is found.

Symptom

Presto cannot query the data written by **union** statements executed by the Tez engine.

Cause Analysis

When Hive uses the Tez engine to execute the **union** statements, the output file is stored in the **HIVE_UNION_SUBDIR** directory. However, Presto does not access files in child directories by default. Therefore, data in the **HIVE_UNION_SUBDIR** directory is not read.

Procedure

Step 1 On the MRS console, click the cluster name, and choose **Components** > **Presto** > **Service Configuration**.

Step 2 Change **Basic** to **All**.

Step 3 In the navigation tree on the left, choose **Presto** > **Hive**. In the **catalog/hive.properties** file, add the **hive.recursive-directories** parameter and set it to **true**.

Step 4 Click **Save Configuration** and select **Restart the affected services or instances**.

----End

16.16 Using Spark

16.16.1 An Error Occurs When the Split Size Is Changed in a Spark Application

Issue

An error occurs when the split size is changed in a Spark application.

Symptom

A user needs to implement multiple mappers by changing the maximum split size to make the Spark application run faster. However, an error occurs when the user runs the **set $Parameter** command to modify the Hive configuration.

```
0: jdbc:hive2://192.168.1.18:21066/> set mapred.max.split.size=1000000;
Error: Error while processing statement: Cannot nodify mapred.max.split.size at runtime. It is not in list of params that are allowed to be modified at runtime( state=42000,code=1)
```

Cause Analysis

- Before the **hive.security.whitelist.switch** parameter is set to enable or disable the whitelist in security mode, the allowed parameters must have been configured in **hive.security.authorization.sqlstd.confwhitelist**.
- The default whitelist does not contain the **mapred.max.split.size** parameter. Therefore, the system displays a message indicating that the maximum split size cannot be changed.

Procedure

Step 1 Search for **hive.security.authorization.sqlstd.confwhitelist.append**, and add **mapred.max.split.size** to **hive.security.authorization.sqlstd.confwhitelist.append**. For details, see **Component Operation Guide** > **Using Hive** > **Using Hive from Scratch**.

Step 2 Save the configuration and restart the Hive component.

Step 3 Run the **set mapred.max.split.size=1000000;** command. If no error occurs, the modification is successful.

----End

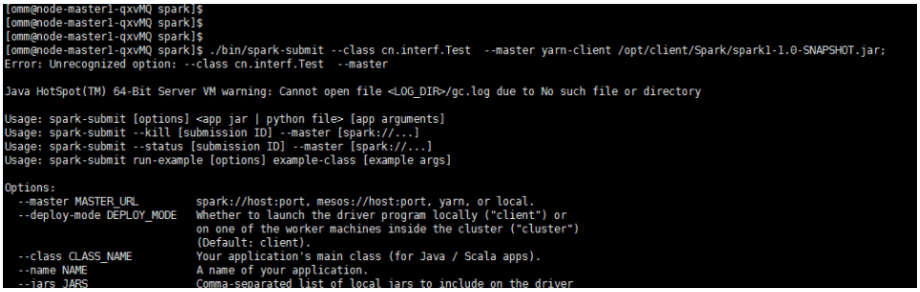
16.16.2 An Error Is Reported When Spark Is Used

Issue

When Spark is used, the cluster fails to run.

Symptom

When Spark is used, the cluster fails to run.



Cause Analysis

- Invalid characters are added during command execution.
- The owner and owner group of the uploaded JAR file is incorrect.

Procedure

Step 1 Run **./bin/spark-submit --class cn.interf.Test --master yarn-client /opt/client/Spark/spark1-1.0-SNAPSHOT.jar;** to check whether invalid characters are imported.

Step 2 If they are imported, modify the invalid characters and run the command again.

Step 3 After the command is executed again, other errors occur. Both the owner and the owner group of the JAR file are **root**.

Step 4 Change the owner and the owner group of the JAR file to **omm:wheel**.

----End

16.16.3 A Spark Job Fails to Run Due to Incorrect JAR File Import

Issue

A Spark job fails to be executed.

Symptom

A Spark job fails to be executed.

Cause Analysis

The imported JAR file is incorrect when the Spark job is executed. As a result, the Spark job fails to be executed.

Procedure

Step 1 Log in to any Master node.

Step 2 Run the **cd /opt/Bigdata/MRS_*/install/FusionInsight-Spark-*/spark/examples/jars** command to view the JAR file of the sample program.

NOTE

A JAR file name contains a maximum of 1023 characters and cannot include special characters (;|&>,<'$). In addition, it cannot be left blank or full of spaces.

Step 3 Check the executable programs in the OBS bucket. The executable programs can be stored in HDFS or OBS. The paths vary according to file systems.

NOTE

- OBS storage path: starts with **obs://**, for example, **obs://wordcount/program/hadoop-mapreduce-examples-2.7.x.jar**.
- HDFS storage path: starts with **/user**. Spark Script must end with **.sql**, and MR and Spark must end with **.jar**. The **.sql** and **.jar** are case-insensitive.

----End

16.16.4 A Spark Job Is Pending Due to Insufficient Memory

Issue

Memory is insufficient to submit a Spark job. As a result, the job is in the pending state for a long time or out of memory (OMM) occurs during job running.

Symptom

The job is pending for a long time after being submitted. The following error information is displayed after the job is executed repeatedly:

```
Exception in thread "main" org.apache.spark.SparkException: Job aborted due to stage failure:
Aborting TaskSet 3.0 because task 0 (partition 0) cannot run anywhere due to node and executor blacklist.
Blacklisting behavior can be configured via spark.blacklist.*.
```

Cause Analysis

The memory is insufficient. As a result, the submitted Spark job is in the pending state for a long time.

Procedure

Step 1 Log in to the MRS console, click a cluster name on the **Active Clusters** page and view the node specifications of the cluster on the **Nodes** tab page.

Step 2 Add cluster resources owned by the **nodemanager** process.

MRS Manager:

1. Log in to MRS Manager and choose **Services** > **Yarn** > **Service Configuration**.
2. Set **Type** to **All**, and then search for **yarn.nodemanager.resource.memory-mb** in the search box to view the value of this parameter. You are advised to set the parameter value to 75% to 90% of the total physical memory of nodes.

FusionInsight Manager:

1. Log in to FusionInsight Manager. Choose **Cluster** > **Service** > **Yarn**.
2. Choose **Configurations** > **All Configurations**. Search for **yarn.nodemanager.resource.memory-mb** in the search box and check the parameter value. You are advised to set the parameter value to 75% to 90% of the total physical memory of nodes.

Step 3 Modify the Spark service configuration.

MRS Manager:

1. Log in to MRS Manager and choose **Services** > **Spark** > **Service Configuration**.
2. Set **Type** to **All**, and then search for **spark.driver.memory** and **spark.executor.memory** in the search box.

   Set these parameters to a larger or smaller value based on the complexity and memory requirements of the submitted Spark job. (Generally, the values need to be increased.)

FusionInsight Manager:

1. Log in to FusionInsight Manager. Choose **Cluster** > **Service** > **Spark**.
2. Choose **Configurations** > **All Configurations**. Search for **spark.driver.memory** and **spark.executor.memory** in the search box and increase or decrease the values based on actual requirements. Generally, increase the values based on the complexity and memory of the submitted Spark job.

NOTE

- If a SparkJDBC job is used, search for **SPARK_EXECUTOR_MEMORY** and **SPARK_DRIVER_MEMORY** and modify their values based on the complexity and memory requirements of the submitted Spark job. (Generally, the values need to be increased.)
- If the number of cores needs to be specified, you can search for **spark.driver.cores** and **spark.executor.cores** and modify their values.

Step 4 Scale out the cluster if the preceding requirements still cannot be met because Spark depends on the memory for computing.

----End

16.16.5 An Error Is Reported During Spark Running

Issue

The specified class cannot be found when a Spark job is running.

Symptom

The specified class cannot be found when a Spark job is running. The error message is as follows:

```
Exception encountered | org.apache.spark.internal.Logging$class.logError(Logging.scala:91)
org.apache.hadoop.hbase.DoNotRetryIOException: java.lang.ClassNotFoundException:
org.apache.phoenix.filter.SingleCQKeyValueComparisonFilter
```

Cause Analysis

The default path configured by the user is incorrect.

Procedure

Step 1 Log in to any Master node.

Step 2 Modify the configuration file in the Spark client directory.

Run the **vim /opt/client/Spark/spark/conf/spark-defaults.conf** command to open the **spark-defaults.conf** file and set **spark.executor.extraClassPath** to **${PWD}/***.

----End

16.16.6 Executor Memory Reaches the Threshold Is Displayed in Driver

Symptom

A Spark task fails to be submitted due to excessive memory usage.

Cause Analysis

The Driver log prints that the applied Executor memory exceeds the cluster limit.

```
16/02/06 14:11:25 INFO Client: Verifying our application has not requested more than the maximum
memory capability of the cluster (6144 MB per container)
16/02/06 14:11:29 ERROR SparkContext: Error initializing SparkContext.
java.lang.IllegalArgumentException: Required executor memory (10240+1024 MB) is above the max
threshold (6144 MB) of this cluster!
```

Spark tasks are submitted to Yarn and the resources used by the Executor to run tasks are managed by Yarn. From the error message, you can see that when a user starts the Executor, 10 GB memory is specified, which exceeds the upper memory limit of each Container set by Yarn. As a result, the task cannot be started.

Solution

Modify the Yarn configuration to increase the restriction on containers. For example, you can adjust parameter **yarn.scheduler.maximum-allocation-mb** to control the resources for starting the Executor. Restart the Yarn service after the modification.

You can modify the configuration as follows:

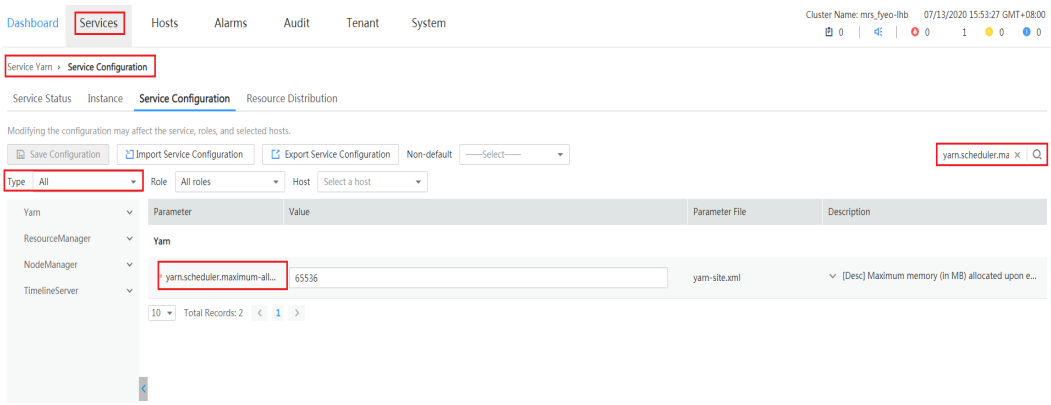
MRS Manager:

Step 1 Log in to MRS Manager.

Step 2 Choose **Services** > **Yarn** > **Service Configuration** and set **Type** to **All**.

Step 3 In **Search**, enter **yarn.scheduler.maximum-allocation-mb** to modify the parameter, save the configuration, and then restart the service. See the following figure.

Figure 16-56 Modifying Yarn service parameters



----End

FusionInsight Manager:

Step 1 Log in to FusionInsight Manager.

Step 2 Choose **Cluster** > **Service** > **Yarn**. Click **Configurations** and select **All Configurations**.

Step 3 In **Search**, enter **yarn.scheduler.maximum-allocation-mb** to modify the parameter, save the configuration, and then restart the service.

----End

16.16.7 Message "Can't get the Kerberos realm" Is Displayed in Yarn-cluster Mode

Symptom

A Spark task fails to be submitted due to an authentication failure.

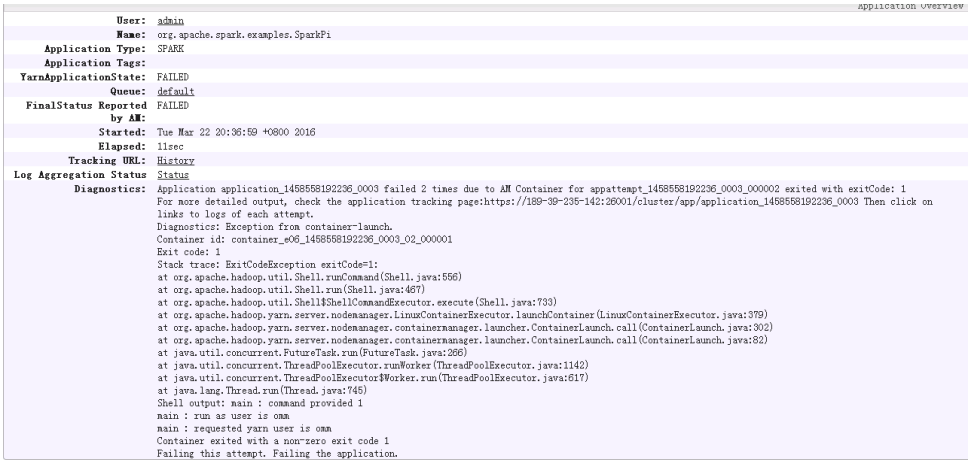
Cause Analysis

1. According to the exception printed in the driver log, the token used to connect to HDFS cannot be found.

```
16/03/22 20:37:10 WARN Client: Exception encountered while connecting to the server :
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.token.SecretManager
$InvalidToken): token (HDFS_DELEGATION_TOKEN token 192 for admin) can't be found in cache
16/03/22 20:37:10 WARN Client: Failed to cleanup staging dir .sparkStaging/
application_1458558192236_0003
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.token.SecretManager
$InvalidToken): token (HDFS_DELEGATION_TOKEN token 192 for admin) can't be found in cache
```

2. The native Yarn web UI shows that ApplicationMaster fails to be started twice and the task exits.

   Figure 16-57 ApplicationMaster start failure



3. The ApplicationMaster log shows the following error information:

```
Exception in thread "main" java.lang.ExceptionInInitializerError
Caused by: org.apache.spark.SparkException: Unable to load YARN support
Caused by: java.lang.IllegalArgumentException: Can't get Kerberos realm
Caused by: java.lang.reflect.InvocationTargetException
Caused by: KrbException: Cannot locate default realm
Caused by: KrbException: Generic error (description in e-text) (60) - Unable to locate Kerberos realm
org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1410)
... 86 more
Caused by: javax.jdo.JDOFatalInternalException: Unexpected exception caught.
NestedThrowables:java.lang.reflect.InvocationTargetException
... 110 more
```

4. When you execute **./spark-submit --class yourclassname --master yarn-cluster /yourdependencyjars** to submit a task in Yarn-cluster mode, the driver is enabled in the cluster. Because the client's **spark.driver.extraJavaOptions** is loaded, you cannot find the **kdc.conf** file in the target path on the cluster node and cannot obtain information required for Kerberos authentication. As a result, the ApplicationMaster fails to be started.

Solution

When submitting a task on the client, configure the **spark.driver.extraJavaOptions** parameter in the CLI. In this way, the **spark.driver.extraJavaOptions** parameter in the **spark-defaults.conf** file is not automatically loaded from the client path. When starting a Spark task, use **--conf** to specify the driver configuration as follows (note that the quotation mark after **spark.driver.extraJavaOptions=** is mandatory):

./spark-submit -class yourclassname --master yarn-cluster --conf spark.driver.extraJavaOptions="

-Dlog4j.configuration=file:/opt/client/Spark/spark/conf/log4j.properties -Djetty.version=x.y.z -Dzookeeper.server.principal=zookeeper/hadoop.794bbab6_9505_44cc_8515_b4eddc84e6c1.com -Djava.security.krb5.conf=/opt/client/KrbClient/kerberos/var/krb5kdc/krb5.conf -Djava.security.auth.login.config=/opt/client/Spark/spark/conf/jaas.conf -Dorg.xerial.snappy.tempdir=/opt/client/Spark/tmp -Dcarbon.properties.filepath=/opt/client/Spark/spark/conf/carbon.properties" ../yourdependencyjars

16.16.8 Failed to Start spark-sql and spark-shell Due to JDK Version Mismatch

Symptom

The JDK version does not match. As a result, the client fails to start spark-sql and spark-shell.

Cause Analysis

1. The following error information is displayed on the Driver:
   Exception Occurs: BadPadding 16/02/22 14:25:38 ERROR Schema: Failed initialising database. Unable to open a test connection to the given database. JDBC url = jdbc:postgresql://ip:port/sparkhivemeta, username = spark. Terminating connection pool (set lazyInit to true if you expect to start your database after your app).
2. When a SparkSQL task is used, DBService needs to be accessed to obtain metadata information. On the client, the ciphertext needs to be decrypted for access. During the use, the user does not follow the process or configure environment variables, and the default JDK version exists in the environment variables of the client. As a result, the decryption program invoked during decryption is abnormal, and the user is locked.

Solution

Step 1 Run the **which java** command to check whether the default Java command is the Java command of the client.

Step 2 If it is not, go to the next step.

source ${client_path}/bigdata_env

Run the **kinit** *username* command and enter the password corresponding to the username to start the task.

----End

16.16.9 ApplicationMaster Failed to Start Twice in Yarn-client Mode

Symptom

In Yarn-client mode, ApplicationMaster fails to start twice.

Cause Analysis

1. Driver exception:

```
16/05/11 18:10:56 INFO Client:
client token: N/A
diagnostics: Application application_1462441251516_0024 failed 2 times due to AM Container for
appattempt_1462441251516_0024_000002 exited with  exitCode: 10
For more detailed output, check the application tracking page:https://hdnode5:26001/cluster/app/
application_1462441251516_0024 Then click on links to logs of each attempt.
Diagnostics: Exception from container-launch.
Container id: container_1462441251516_0024_02_000001
```

2. The ApplicationMaster log file contains the following error information:

```
2016-05-12 10:21:23,715 | ERROR | [main] | Failed to connect to driver at 192.168.30.57:23867,
retrying ... | org.apache.spark.Logging$class.logError(Logging.scala:75)
2016-05-12 10:21:24,817 | ERROR | [main] | Failed to connect to driver at 192.168.30.57:23867,
retrying ... | org.apache.spark.Logging$class.logError(Logging.scala:75)
2016-05-12 10:21:24,918 | ERROR | [main] | Uncaught exception:  | org.apache.spark.Logging
$class.logError(Logging.scala:96)
org.apache.spark.SparkException: Failed to connect to driver!
at org.apache.spark.deploy.yarn.ApplicationMaster.waitForSparkDriver(ApplicationMaster.scala:426)
at org.apache.spark.deploy.yarn.ApplicationMaster.runExecutorLauncher(ApplicationMaster.scala:292)
...
2016-05-12 10:21:24,925 | INFO  | [Thread-1] | Unregistering ApplicationMaster with FAILED (diag
message: Uncaught exception: org.apache.spark.SparkException: Failed to connect to driver!) |
org.apache.spark.Logging$class.logInfo(Logging.scala:59)
```

In Spark-client mode, the task Driver runs on a client node (usually a node outside the cluster). During the startup, the ApplicationMaster process is started in the cluster. After the process is started, information needs to be registered with the Driver process. The task can be continued only after the registration is successful. According to the ApplicationMaster log, the connection to the Driver fails, which causes the task failure.

Solution

Step 1 Check whether the IP address of the Driver process can be pinged.

Step 2 Start a SparkPI task. Information similar to the following is displayed on the console:

```
16/05/11 18:07:20 INFO Remoting: Remoting started; listening on addresses :[akka.tcp://
sparkDriver@192.168.1.100:23662]
16/05/11 18:07:20 INFO Utils: Successfully started service 'sparkDriver' on port 23662.
```

Step 3 Run the **netstat - anp | grep 23662** command on the node (192.168.1.100 in **Step 2**) to check whether the port is enabled. The following information indicates that the port is enabled.

```
tcp       0      0  ip:port    :::*                    LISTEN      107274/java
tcp       0      0  ip:port    ip:port                 ESTABLISHED 107274/java
```

Step 4 Run the **telnet 192.168.1.100 23662** command on the node where ApplicationMaster is started to check whether the port can be connected. Perform this operation as both the **root** and **omm** users. If information similar to **Escape character is '^]'** is displayed, the connection is normal. If **connection refused** is displayed, the connection fails and the related port cannot be connected.

If the port is enabled but cannot be connected from other nodes, check the network configuration.

NOTE

The port (port 23662 in this example) is randomly selected each time. Therefore, you need to test the port enabled by the task.

----End

16.16.10 Failed to Connect to ResourceManager When a Spark Task Is Submitted

Symptom

The connection to ResourceManager is abnormal. As a result, Spark tasks fail to be submitted.

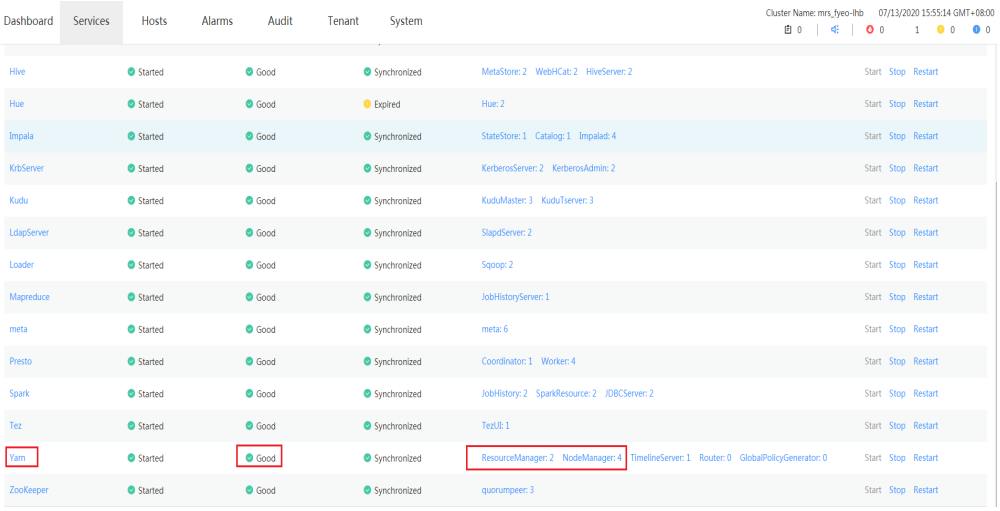
Cause Analysis

1. The following error information is displayed on the Driver, indicating that port 26004 connecting to the active and standby ResourceManager nodes is rejected:

```
15/08/19 18:36:16 INFO RetryInvocationHandler: Exception while invoking getClusterMetrics of class ApplicationClientProtocolPBClientImpl over 33 after 1 fail over attempts. Trying to fail over after sleeping for 17448ms.
 java.net.ConnectException: Call From ip0 to ip1:26004 failed on connection exception: java.net.ConnectException: Connection refused.
INFO RetryInvocationHandler: Exception while invoking getClusterMetrics of class ApplicationClientProtocolPBClientImpl over 32 after 2 fail over attempts. Trying to fail over after sleeping for 16233ms.
 java.net.ConnectException: Call From ip0 to ip2:26004 failed on connection exception: java.net.ConnectException: Connection refused;
```

2. On MRS Manager, check whether ResourceManager is running properly, as shown in **Figure 16-58**. If Yarn is faulty or an unknown exception occurs on a Yarn service instance, ResourceManager of the cluster may be abnormal.

Figure 16-58 Service status



3. Check whether the client is the latest one in the cluster.

   Check whether the ResourceManager instance has been migrated in the cluster. (Uninstall a ResourceManager instance and add it back to other nodes.)

4. On MRS Manager, click **Audit** to view audit logs and check whether related operations are recorded.

   Run the **ping** command to check whether the IP address can be pinged.

Solution

- If ResourceManager is abnormal, see the Yarn-related sections to rectify the fault.
- If the client is not the latest, download the client again.
- If the IP address cannot be pinged, contact network management personnel to check the network.

16.16.11 DataArts Studio Failed to Schedule Spark Jobs

Issue

DataArts Studio fails to schedule jobs, and a message is displayed indicating that data in the **/thriftserver/active_thriftserver** directory cannot be read.

Symptom

DataArts Studio fails to schedule jobs, and the following error is reported indicating that data in the **/thriftserver/active_thriftserver** directory cannot be read:

Can not get JDBC Connection, due to KeeperErrorCode = NoNode for /thriftserver/active_thriftserver

Cause Analysis

When DataArts Studio submits a Spark job, Spark JDBC is invoked. Spark starts a ThriftServer process for the client to provide JDBC connections. During the startup, JDBCServer creates the **active_thriftserver** subdirectory in the **/thriftserver** directory of ZooKeeper, and registers related connection information. If the connection information cannot be read, the JDBC connection is abnormal.

Procedure

Check whether the ZooKeeper directory contains the target directory and registration information.

Step 1 Log in to any master node as user **root** and initialize environment variables.

source /opt/client/bigdata_env

Step 2 Run the **zkCli.sh -server 'ZookeeperIp:2181'** command to log in to ZooKeeper.

Step 3 Run the **ls /thriftserver** command to check whether the **active_thriftserver** directory exists.

- If the **active_thriftserver** directory exists, run the **get /thriftserver/active_thriftserver** command to check whether it contains the registered configuration information.
  - If yes, contact technical support.

  - If no, go to **Step 4**.
- If the **active_thriftserver** directory does not exist, go to **Step 4**.

Step 4 Log in to Manager and check whether the active/standby status of the Spark JDBCServer instance is unknown.

- If yes, go to **Step 5**.
- If no, contact O&M personnel.

Step 5 Restart the two JDBCServer instances. Check whether the status of the active and standby instances is normal and whether the target directory and data exist in ZooKeeper. If yes, the job is restored. If the instance status is not restored, contact technical support.

----End

16.16.12 Submission Status of the Spark Job API Is Error

Issue

After a Spark job is submitted using an API, the job status is displayed as **error**.

Issue Type

Job management

Symptom

After the log level in **/opt/client/Spark/spark/conf/log4j.properties** is changed and a job is submitted using API V1.1, the job status is displayed as error.

Cause Analysis

The executor monitors the job log output and determines the job execution result. After the execution result is changed to **error**, the output result cannot be detected. Therefore, the executor determines that the job status is abnormal after the job expires.

Procedure

Change the log level in the **/opt/client/Spark/spark/conf/log4j.properties** file to **info**.

Summary and Suggestions

You are advised to use the V2 API to submit jobs.

16.16.13 Alarm 43006 Is Repeatedly Generated in the Cluster

Issue

The alarm "ALM-43006 Heap Memory Usage of the JobHistory Process Exceeds the Threshold" is repeatedly generated in the cluster, and the setting according to the alarm reference is invalid.

Symptom

Alarm **ALM-43006 Heap Memory Usage of the JobHistory Process Exceeds the Threshold** is generated in the cluster. The same alarm is generated again a period of time after handling measures are taken.

Cause Analysis

The JobHistory memory leakage may occur. You need to install the corresponding patch to rectify the fault.

Procedure

- Increase the heap memory of the JobHistory process.
- If the heap memory has been increased, restart the JobHistory instance.

16.16.14 Failed to Create or Delete a Table in Spark Beeline

Issue

When the customer frequently creates or deletes a large number of users in Spark Beeline, some users occasionally fail to create or delete tables.

Symptom

The procedure for creating a table is as follows:

```
CREATE TABLE wlg_test001 (start_time STRING,value INT);
```

The following error message is displayed:

```
Error: org.apache.spark.sql.AnalysisException:
org.apache.hadoop.hive.ql.metadata.HiveException: MetaException(message:Failed to grant permission on
HDFSjava.lang.reflect.UndeclaredThrowableException); (state=,code=0)
```

Cause Analysis

1. View metastore logs.

2. View HDFS logs.

```
2020-08-31 14:41:38,568 | INFO  | Socket Reader #1 for port 9820 | Authorization successful for hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com@036A3461_D09B_494F_A32C_AF273307D943.COM (auth:KERBEROS) for protocol=interface org.apache.hadoop.hdfs.protocol.ClientProtocol | ServiceAuthorizationManager.java:135
2020-08-31 14:41:38,586 | INFO  | IPC Server handler 7 on 9820 | IPC Server handler 7 on 9820, call Call#3822197 Retry#0 org.apache.hadoop.hdfs.protocol.ClientProtocol.checkAccess from 192.168.1.66:50540: org.apache.hadoop.security.AccessControlException: Permission denied: user=hive, access=READ, inode="/user/hive/warehouse/hive_csb_csb_3f8_x48ssrbt_5lbi2edu.db/wlg_test001":spark:hive:drwx------ | Server.java:2523
2020-08-31 14:41:38,852 | INFO  | Socket Reader #1 for port 9820 | Auth successful for hwstaff_pub_0tw00ru6@036A3461_D09B_494F_A32C_AF273307D943.COM (auth:TOKEN) | Server.java:1700
2020-08-31 14:41:38,911 | INFO  | Socket Reader #1 for port 9820 | Authorization successful for hwstaff_pub_0tw00ru6@036A3461_D09B
```

3. Compare permission (**test001** is a table created by a user in abnormal state, and **test002** is a table created by a user in normal state).

```
drwx------   - spark               hive          0 2020-08-31 14:41 /user/hive/warehouse/hive_csb_csb_3f8_x48ssrbt_5lbi2edu.db/wlg_test001
drwxrwx---+  - spark               hive          0 2020-08-31 15:07 /user/hive/warehouse/hive_csb_csb_3f8_x48ssrbt_5lbi2edu.db/wlg_test002
```

4. An error similar to the following is reported when a table is dropped:

```
0: jdbc:hive2://192.168.1.42:10000/> drop table
dataplan_modela_csbch2;
Error: Error while compiling statement: FAILED:
SemanticException Unable to fetch table dataplan_modela_csbch2.
java.security.AccessControlException: Permission denied: user=CSB_csb_3f8_x48ssrbt,
access=READ,
inode="/user/hive/warehouse/hive_csb_csb_3f8_x48ssrbt_5lbi2edu.db/
dataplan_modela_csbch2":spark:hive:drwx------
```

5. Analyze the cause.

   The default user created during cluster creation uses the same UID, causing user disorder. This problem is triggered when a large number of users are created. As a result, the Hive user does not have the permission to create tables occasionally.

```
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]# id hive
uid=20013(hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com) gid=10002(hive) groups=10002(hive)
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]# id hive
uid=20013(hive) gid=10002(hive) groups=10002(hive),10001(hadoop),10000(supergroup),8003(System_administrator_186),9998(ficommon)
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]#
[root@node-master2iMrt ~]#
```

```
objectClass: krbPrincipalAux
objectClass: krbTicketPolicyAux

# hive, Peoples, hadoop.com
dn: cn=hive,ou=Peoples,dc=hadoop,dc=com
uid: hive
homeDirectory: /home/hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com
cn: hive
uidNumber: 20013
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
userPassword:: e1NTSEF9cXZWS0VlMi9pYVFpdzFmUmNIUVJFUEJYZWtKLzZHMHk=
gidNumber: 10002

# hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com, Peoples, hadoop.com
dn: cn=hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com,ou=Peoples,dc=hadoop,dc=com
uid: hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com
homeDirectory: /home/hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com
cn: hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com
uidNumber: 20013
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
gidNumber: 10002
description: [userName:"hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com"]
description: [userType:"1"]
description: [groupList:"hive,hadoop,supergroup,compcommon"]
description: [roleList:"System_administrator"]
description: [description:"aGl2ZSBkZWZhdWx0IHVzZXIjSGl2Zem7mOiupOeUqOaItw=="]
description: [createTime:"1554974652422"]
description: [defaultUser:"0"]
description: [primaryGroup:"hive"]

# hive/hadoop.036a3461_d09b_494f_a32c_af273307d943.com@036A3461_D09B_494F_A32C_AF273307D943.COM, 036A3461_D09B_494F_A32C_AF273307D943.COM, krbcontainer, hado
```

Procedure

Restart the **sssd** process of the cluster.

Run the **service sssd restart** command as the **root** user to restart the **sssd** process and run the **ps -ef | grep sssd** command to check whether the **sssd** process is running properly.

In normal cases, the **/usr/sbin/sssd** process and three sub-processes **/usr/libexec/sssd/sssd_be**, **/usr/libexec/sssd/sssd_nss** and **/usr/libexec/sssd/sssd_pam** exist.

16.16.15 Failed to Connect to the Driver When a Node Outside the Cluster Submits a Spark Job to Yarn

Issue

When a node outside the cluster uses the client mode to submit a Spark task to Yarn, the task fails and an error message is displayed, indicating that the driver cannot be connected.

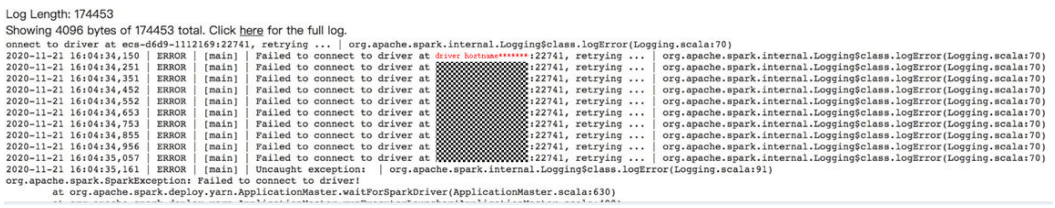
Symptom

Nodes outside the cluster can communicate with each node in the cluster. When a node outside the cluster submits a Spark task to Yarn in client mode, the task fails and an error message is displayed, indicating that the driver cannot be connected.

Cause Analysis

When a Spark task is submitted in the client mode, the driver process of Spark is on the client side, and the executor needs to interact with the driver to run the job.

If the NodeManager fails to connect to the node where the client is located, the following error is reported:



Procedure

Specify the IP address of the driver in the Spark configuration of the client.

Add **spark.driver.host=driverIP** to *<Client installation path>***/Spark/spark/conf/spark-defaults.conf** and run the Spark task again.

Summary and Suggestions

You are advised to submit jobs in cluster mode.

16.16.16 Large Number of Shuffle Results Are Lost During Spark Task Execution

Issue

Spark tasks fail to be executed. The task log shows that a large number of **shuffle** files are lost.

Symptom

Spark tasks fail to be executed. The task log shows that a large number of **shuffle** files are lost.

Cause Analysis

When Spark is running, the **shuffle** file generated temporarily is stored in the temporary directory of the executor for later use.

When an executor exits abnormally, NodeManager deletes the temporary directory of the container where the executor is located. When other executors apply for the shuffle result of the executor, a message is displayed indicating that the file cannot be found.

Therefore, you need to check whether the executor exits abnormally. You can check whether there are executors in the **dead** state on the executors tab page on the Spark task page and view the executor logs of each **dead** state, determine the cause of abnormal exit. Some executors may exit because the **shuffle** file cannot be found. You need to find the earliest executor that exits abnormally.

Common abnormal exit causes:

- OOM occurs on the executor.
- Multiple tasks fail when the executor is running.
- The node where the executor is located is cleared.

Procedure

Adjust or modify the task parameters or code based on the actual cause of the abnormal exit of the executor, and run the Spark task again.

16.16.17 Disk Space Is Insufficient Due to Long-Term Running of JDBCServer

Issue

When the JDBCServer service connected to Spark submits a spark-sql task to the Yarn cluster, the data disk of the Core node is fully occupied after the task runs for a period of time.

Symptom

When the JDBCServer service of a customer connected to Spark submits a spark-sql task to the Yarn cluster, the data disk of the Core node is fully occupied after the task runs for a period of time.

After checking the disk usage in the background, it is found that there are too many APP temporary files (files generated by shuffle) of the JDBCServer service, and the files are not cleared, occupying a large amount of memory.

Cause Analysis

After checking the directories that contain a large number of files on the Core node, it is found that most of the directories are similar to **blockmgr-033707b6-fbbb-45b4-8e3a-128c9bcfa4bf**, which stores temporary shuffle files generated during computing.

The dynamic resource allocation function of Spark is enabled on JDBCServer, and shuffle is hosted by NodeManager. NodeManager only manages these files based on the running period of the application, and does not check whether the container where a single executor is located exists. Therefore, the temporary files are deleted only when the app is stopped. When a task runs for a long time, a large number of temporary files occupy a large amount of disk space.

Procedure

Start a scheduled task to delete shuffle files that have been stored for a specified period of time. For example, delete shuffle files that have been stored for more than 6 hours each hour.

Step 1 Create the **clean_appcache.sh** script. If there are multiple data disks, change the value of **data1** in **BASE_LOC** based on the actual situation.

- Security cluster

```
#!/bin/bash
BASE_LOC=/srv/BigData/hadoop/data1/nm/localdir/usercache/spark/appcache/application_*/blockmgr*
find $BASE_LOC/ -mmin +360 -exec rmdir {} \;
find $BASE_LOC/ -mmin +360 -exec rm {} \;
```

- Common cluster

```
#!/bin/bash
BASE_LOC=/srv/BigData/hadoop/data1/nm/localdir/usercache/omm/appcache/application_*/blockmgr*
find $BASE_LOC/ -mmin +360 -exec rmdir {} \;
find $BASE_LOC/ -mmin +360 -exec rm {} \;
```

Step 2 Run the following commands to change the permission to the script:

chmod 755 clean_appcache.sh

Step 3 Add a scheduled task to start the clearance script. Change the script path to the actual path.

Run the **crontab -l** command to view the scheduled task.

Run the **crontab -e** command to edit the scheduled task.

```
0 * * * * sh /root/clean_appcache.sh > /dev/null 2>&1
```

----End

16.16.18 Failed to Load Data to a Hive Table Across File Systems by Running SQL Statements Using Spark Shell

Issue

When the **spark-shell** command is used to execute SQL statements or the **spark-submit** command is used to submit Spark tasks, the **load** command of SQL statements exists, and the source data and target table are not stored in the same file system. An error is reported when the MapReduce task is started in the preceding two modes.

Cause Analysis

When the **load** command is used to import data to the Hive table across file systems (for example, the original data is stored in the HDFS but the Hive table data is stored in the OBS), and the file length is greater than the threshold (32 MB by default). In this case, the MapReduce job that uses DistCp is triggered to migrate data. The MapReduce task configuration is directly extracted from the Spark task configuration. However, the **net.topology.node.switch.mapping.impl** configuration item of the Spark task does not retain the default value of the Hadoop. Therefore, the JAR package of the Spark needs to be used. As a result, the MapReduce reports an error indicating that the class cannot be found.

Procedure

Solution 1:

If the file size is small, set the default file size to a value greater than the maximum file size. For example, if the maximum file size is 95 MB, run the following command:

```
hive.exec.copyfile.maxsize=104857600
```

Solution 2:

If the file size is large, use DistCp to improve the data migration efficiency. Add the following parameters when starting the Spark task:

```
--conf spark.hadoop.net.topology.node.switch.mapping.impl=org.apache.hadoop.net.ScriptBasedMapping
```

16.16.19 Spark Task Submission Failure

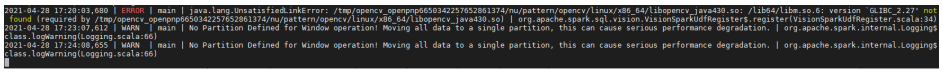
Symptom

- A Spark task fails to be submitted.
- Spark displays a message indicating that the Yarn JAR package cannot be obtained.
- A file is submitted for multiple times.

Cause Analysis

- Symptom 1:

  The most common cause for task submission failure is authentication failure.



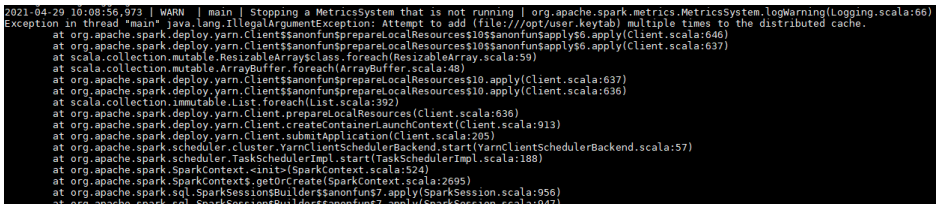
The parameter settings may be incorrect.

- Symptom 2:

  By default, the cluster adds the Hadoop JAR package of the analysis node to the classpath of the task. If the system displays a message indicating that Yarn packages cannot be found, the Hadoop configuration is not set.

- Symptom 3:

  The common scenario is as follows: The **--files** option is used to upload the **user.keytab** file, and then the **--keytab** option is used to specify the same file. As a result, the same file is uploaded for multiple times.



Procedure

- Symptom 1:

  Run **kinit** *[user]* again and modify the corresponding configuration items.

- Symptom 2:

  Check that the Hadoop configuration items are correct and the **core-site.xml**, **hdfs-site.xml**, **yarn-site.xml**, and **mapred-site.xml** configuration files in the **conf** directory of Spark are correct.

- Symptom 3:

  Copy a new **user.keytab** file, for example:

  cp user.keytab user2.keytab

  spark-submit --master yarn --files user.keytab --keytab user2.keytab .....

16.16.20 Spark Task Execution Failure

Symptom

- An executor out of memory (OOM) error occurs.
- The information about the failed task shows that the failure cause is "lost task xxx."

Cause Analysis

- Symptom 1: The data volume is too large or too many tasks are running on the same executor at the same time.
- Symptom 2: Some tasks fail to be executed. When the error is reported, determine the node where the lost task is running. Generally, the error is caused by the abnormal exit of the lost task.

Procedure

- Symptom 1:
  - If the data volume is too large, adjust the memory size of the executor and use **--executor-memory** to specify the memory size.
  - If too many tasks are running at the same time, check the number of vcores specified by **--executor-cores**.
- Symptom 2: Locate the cause in the corresponding task log. If an OOM error occurs, see the solutions to symptom 1.

16.16.21 JDBCServer Connection Failure

Symptom

- The ha-cluster cannot be identified (unknowHost or port required).
- Failed to connect to JDBCServer.

Cause Analysis

- Symptom 1: The **spark-beeline** command is used to connect to JDBCServer. JDBCServer in versions earlier than MRS_3.0 adopts HA mode. Therefore, a specific URL and the JAR package provided by MRS Spark is required to connect to JDBCServer.
- Symptom 2: The JDBCServer service is not running properly or port listening is abnormal.

Procedure

- Symptom 1: Use a specific URL and the JAR package provided by MRS Spark to connect to JDBCServer.
- Symptom 2: Check that the JDBCServer service is running properly and port listening is normal, and try again.

16.16.22 Failed to View Spark Task Logs

Symptom

- A user fails to view logs when a task is running.
- A user fails to view logs when a task is complete.

Cause Analysis

- Symptom 1: The MapReduce component is abnormal.
- Symptom 2:
  - The JobHistory service of Spark is abnormal.
  - The log size is too large, and NodeManager times out during log aggregation.
  - The permission on the HDFS log storage directory (**/tmp/logs/Username/logs** by default) is abnormal.

- Logs have been deleted. By default, Spark JobHistory stores event logs for seven days (specified by **spark.history.fs.cleaner.maxAge**). MapReduce stores task logs for 15 days (specified by **mapreduce.jobhistory.max-age-ms**).
- If the task cannot be found on the Yarn page, it may have been cleared by Yarn. By default, Yarn stores 10,000 historical tasks (specified by **yarn.resourcemanager.max-completed-applications**).

Procedure

- Symptom 1: Check whether the MapReduce component is running properly. If it is abnormal, restart it. If the fault persists, check the JobhistoryServer log file in the background.
- Symptom 2: Perform the following checks in sequence:
  a. Check whether JobHistory of Spark is running properly.
  b. On the app details page of Yarn, check whether the log file is too large. If log aggregation fails, the value of **Log Aggregation Status** should be **Failed** or **Timeout**.
  c. Check whether the permission on the corresponding directory is normal.
  d. Check whether the corresponding **appid** file exists in the directory. In MRS 3.x or later, the event log files are stored in the **hdfs://hacluster/spark2xJobHistory2x** directory. In versions earlier than MRS 3.x, the event log files are stored in the **hdfs://hacluster/sparkJobHistory** directory. The task run logs are stored in the **hdfs://hacluster/tmp/logs/*Username*/logs** directory.
  e. Check whether **appid** or the current job ID exceeds the maximum value in the historical records.

16.16.23 Authentication Fails When Spark Connects to Other Services

Symptom

- When Spark connects to HBase, an authentication failure message is displayed or the HBase table cannot be connected.
- When Spark connects to HBase, a message is displayed indicating that the JAR package cannot be found.

Cause Analysis

- Symptom 1: HBase does not obtain the authentication information of the current task. As a result, the authentication fails when HBase is connected, and the corresponding data cannot be read
- Symptom 2: By default, Spark does not load the HBase JAR package. You need to use **--jars** to add the JAR package to the task.

Procedure

- Symptom 1: Enable the HBase authentication function by running the **spark.yarn.security.credentials.hbase.enabled=true** command. However, do

not replace **hbase-site.xml** on the Spark client with **hbase-site.xml** on the HBase client because they are not completely consistent.

- Symptom 2: Use **--jars** to upload the HBase JAR package.

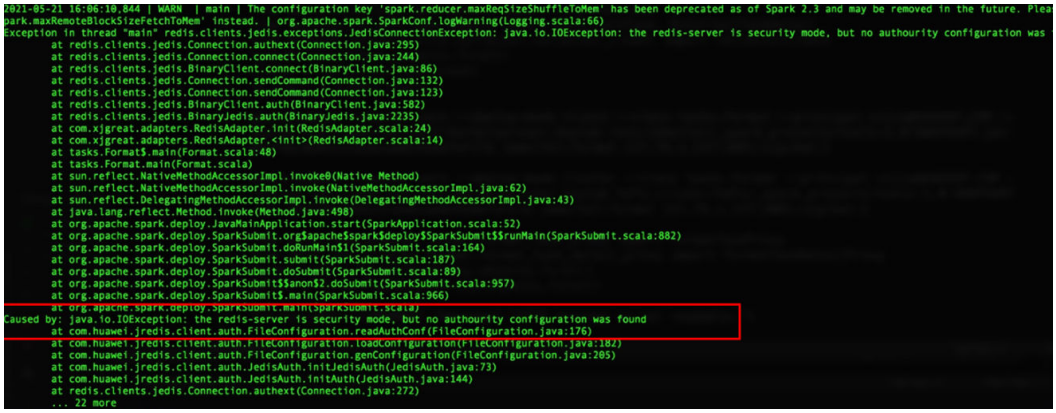
16.16.24 An Error Occurs When Spark Connects to Redis

Issue

An error occurs when the Spark component of the MRS 3.x security cluster is used to access Redis.

Symptom

When Spark of the MRS 3.0 security cluster is used to access Redis, the following error message is displayed.



Cause Analysis

The **jars** directory of Spark contains a **jredisclient-xxx.jar** package provided by the MRS cluster. This package is loaded when a Spark task connects to Redis, thereby causing this error. You can manually remove this package to rectify the fault.

Procedure

Step 1 Delete JAR packages from the Spark client.

cd $SPARK_HOME/jars

mv jredisclient-*.jar /tmp

Step 2 Delete JAR packages from the Spark server.

Log in to the nodes (generally two) where SparkResource2x is located.

mkdir /tmp/SparkResource2x

cd /opt/Bigdata/FusionInsight_Current/1_*_SparkResource2x/install/spark/jars/

mv jredisclient-*.jar /tmp/SparkResource2x

Step 3 Delete the **jredisclient** file from the HDFS.

1. Check configuration item **spark.yarn.archive** in the **$SPARK_HOME/conf/spark-defaults.conf** file to obtain the address of the **spark-archive-2x.zip** package.

   cat $SPARK_HOME/conf/spark-defaults.conf | grep "spark.yarn.archive"

2. Download the **spark-archive-2x.zip** package. (This section uses MRS 3.0.5 as an example. Modify the command based on the actual cluster version.)

   cd /opt

   mkdir sparkTmp

   cd sparkTmp

   hdfs dfs -get hdfs://hacluster/user/spark2x/jars/8.0.2.1/spark-archive-2x.zip

3. Decompress **spark-archive-2x.zip** and remove the package file.

   unzip spark-archive-2x.zip

   rm -f spark-archive-2x.zip

4. Remove the **jredisclient** package.

   rm -f jredisclient-*.jar

5. Compress the **spark-archive-2x.zip** package again.

   zip spark-archive-2x.zip ./*

6. Back up the original package from the HDFS to **tmp** and upload the newly compressed package to the HDFS.

   hdfs dfs -mv hdfs://hacluster/user/spark2x/jars/8.0.2.1/spark-archive-2x.zip /tmp

   hdfs dfs -put spark-archive-2x.zip hdfs://hacluster/user/spark2x/jars/8.0.2.1/spark-archive-2x.zip

7. Restart the JDBCServer service to prevent JDBCServer exceptions. The **jredisclient** file has been deleted from the **spark-archive-2x.zip** package.

8. Delete temporary files.

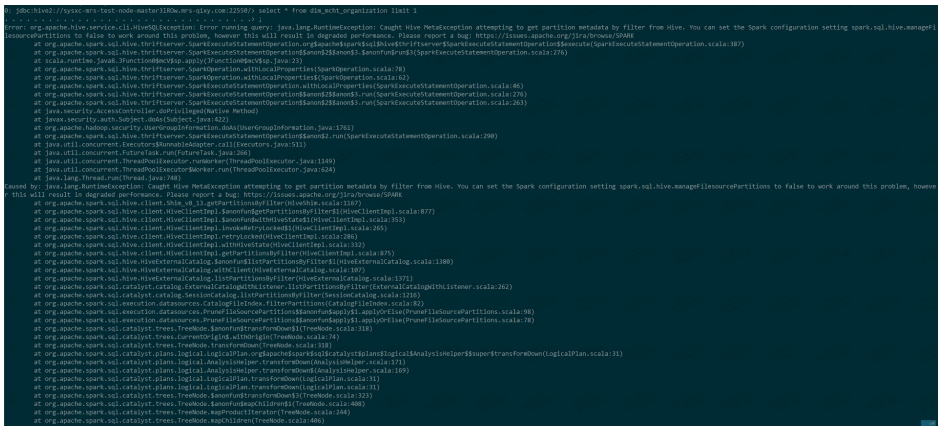
   rm -rf /opt/sparkTmp

----End

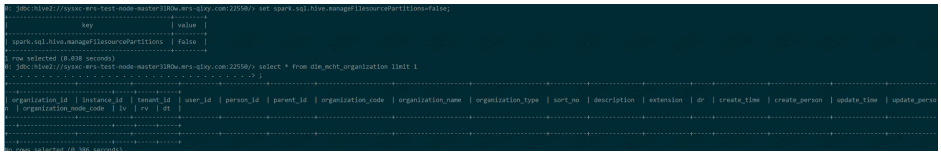
16.16.25 An Error Is Reported When spark-beeline Is Used to Query a Hive View

Issue

In MRS 3.1.2, an error is reported when spark-beeline is used to query a Hive view. The error information is as follows.



After **spark.sql.hive.manageFilesourcePartitions=false** is set as prompted, no data can be found. However, data can be queried in Hive.



Cause Analysis

Failed to convert the format. Spark SQL uses its built-in Metastore instead of Hive Metastore. As a result, metadata fails to be read. Spark SQL reads data in Parquet format, but Hive reads data in ORC format by default.

Procedure

Step 1 Log in to the Spark client node and run the following commands to access Spark SQL:

cd *Client installation directory*

source bigdata_env

source Spark2x/component_env

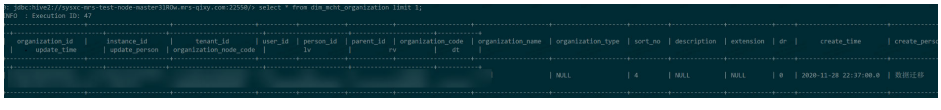
kinit *Component service user* (**kinit** is not required in a normal cluster.)

spark-sql

Step 2 Run the following command to set **spark.sql.hive.convertMetastoreOrc** to **false**:

set spark.sql.hive.convertMetastoreOrc=false;

Step 3 Query the Hive view again.



----End

16.17 Using Sqoop

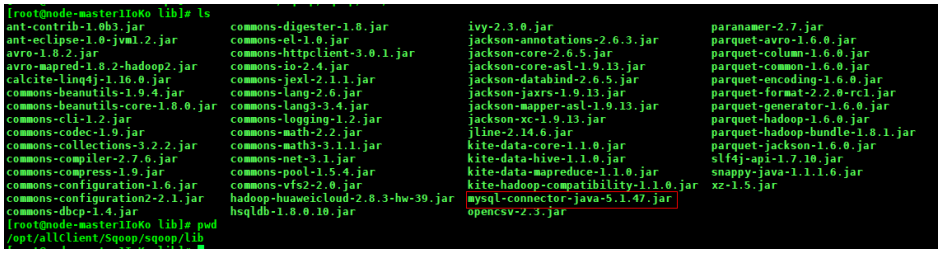
16.17.1 Connecting Sqoop to MySQL

Issue

The user does not know how to connect Sqoop to MySQL.

Procedure

Step 1 Install the client in the cluster and check whether the MySQL driver package exists in the **sqoop/lib** directory of the client.



Step 2 Load environment variables in the client directory.

source bigdata_env

Step 3 Perform the Kerberos authentication.

If Kerberos authentication is not enabled for the cluster, skip this step. If it is enabled, run the following command to authenticate the current user:

kinit *MRS cluster user*

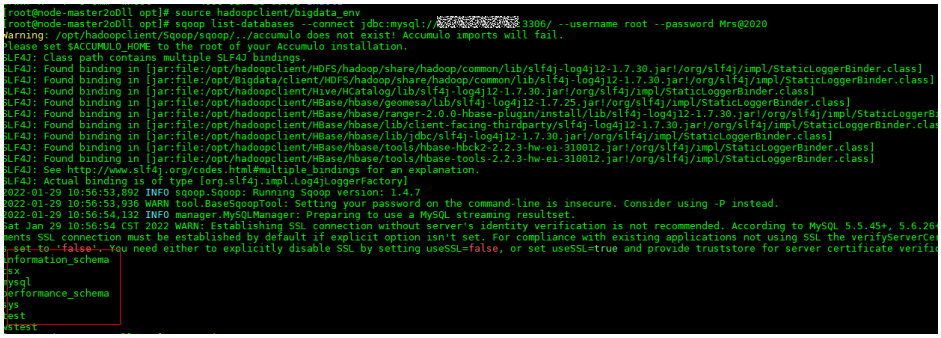
For example:

kinit admin

Step 4 Connect to the database.

sqoop list-databases --connect jdbc:mysql://IP:3306/ --username ***Username*** **--password** ***Password***

An example is as follows.



The command output shows that Sqoop is successfully connected to the MySQL database.

----End

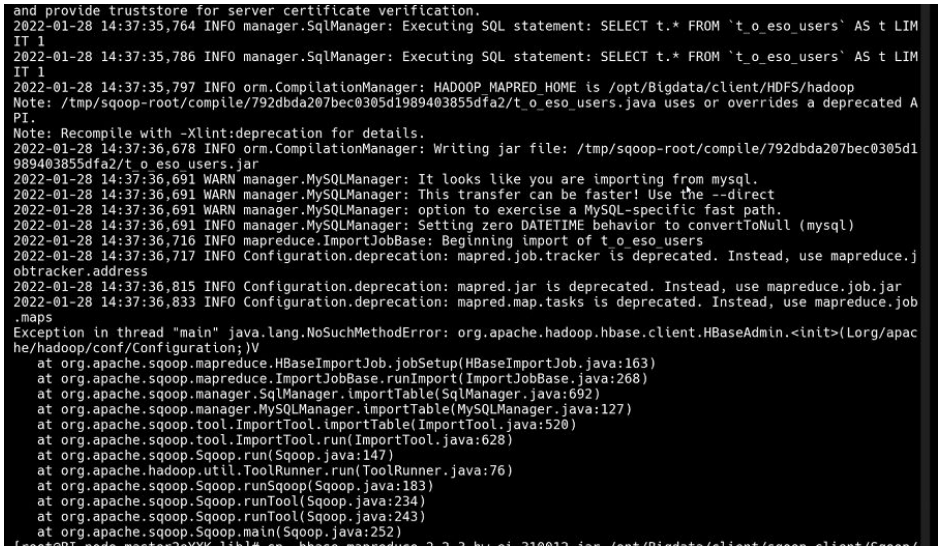
16.17.2 Failed to Find the HBaseAdmin.<init> Method When Sqoop Reads Data from the MySQL Database to HBase

Issue

If the Sqoop client (version 1.4.7) of MRS is used to extract data from a specified table in the MySQL database to a table in HBase 2.2.3, the following exception is reported:

```
Trying to load data into HBASE through Sqoop getting below error.
Exception in thread "main" java.lang.NoSuchMethodError:
org.apache.hadoop.hbase.client.HBaseAdmin.<init>(Lorg/apache/hadoop/conf/Configuration;)V
```

The following figure shows the complete exception information.



The following is an example of running the Sqoop command to extract data:

```
sqoop import \
--connect jdbc:mysql://mysql<i>Server address:Port number</i>/database1 \
--username admin \
--password <i>xxx</i> \
--table table1 \
--hbase-table table2 \
--column-family info \
--hbase-row-key id \
--hbase-create-table --m 1
```

Procedure

After the Sqoop client is installed, the JAR packages on which HBase depends are not imported. You need to manually import the JAR packages on which HBase of an earlier version depends.

Step 1 Check whether the Sqoop and HBase clients are in the same path.

- If yes, go to **Step 2**.
- If no, delete the original Sqoop and HBase client files, download the complete clients from FusionInsight Manager, and install them in the same path. Then go to **Step 2**.

Step 2 Log in to the node where the Sqoop client is installed as user **root**.

Step 3 Download JAR packages of HBase 1.6.0 and upload them to the **lib** directory on the Sqoop client:

Step 4 After the packages are uploaded, run the following command to change the permission on the packages to **755**:

chmod 755 *Package name*

Step 5 Run the following command in the client directory to refresh the Sqoop client:

source bigdata_env

Run the target Sqoop command again.

----End

16.17.3 Failed to Export HBase Data to HDFS Through Hue's Sqoop Task

This section applies only to MRS 1.9.2 clusters.

Issue

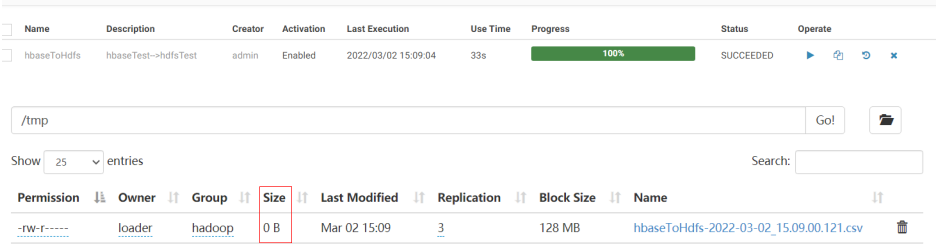
An error is reported when a Sqoop operation is performed on Hue to export data from HBase to HDFS.

Caused by: java.lang.ClassNotFoundException: org.apache.htrace.Trace

```
2022-03-02 15:09:00,264 [main] ERROR org.apache.sqoop.connector.hbase.HBaseExtractor  - An exceptional condition has occurred.
org.apache.sqoop.common.SqoopException: HBASE_CONNECTOR_0011:Failed to open table.
        at org.apache.sqoop.connector.hbase.HBaseExtractor.openDB(HBaseExtractor.java:239)
        at org.apache.sqoop.connector.hbase.HBaseExtractor.access$100(HBaseExtractor.java:34)
        at org.apache.sqoop.connector.hbase.HBaseExtractor$1.run(HBaseExtractor.java:86)
        at org.apache.sqoop.connector.hbase.HBaseExtractor$1.run(HBaseExtractor.java:76)
        at org.apache.sqoop.connector.hbase.HBaseExtractor.extract(HBaseExtractor.java:114)
        at org.apache.sqoop.connector.hbase.HBaseExtractor.extract(HBaseExtractor.java:34)
        at org.apache.sqoop.job.mr.SqoopMapper.runInternal(SqoopMapper.java:156)
        at org.apache.sqoop.job.mr.SqoopMapper.run(SqoopMapper.java:79)
        at org.apache.hadoop.mapred.MapTask.runNewMapper(MapTask.java:787)
        at org.apache.hadoop.mapred.MapTask.run(MapTask.java:341)
        at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:188)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1840)
        at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:182)
Caused by: java.lang.reflect.InvocationTargetException
```

Symptom

The Sqoop task is executed successfully, but the CSV file in HDFS is empty.



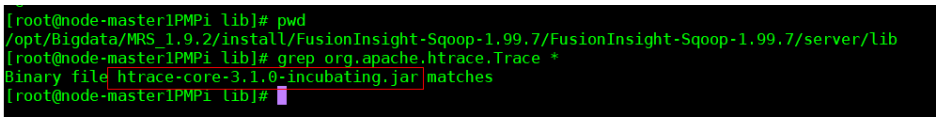
Cause Analysis

The JAR package conflicts or is missing.

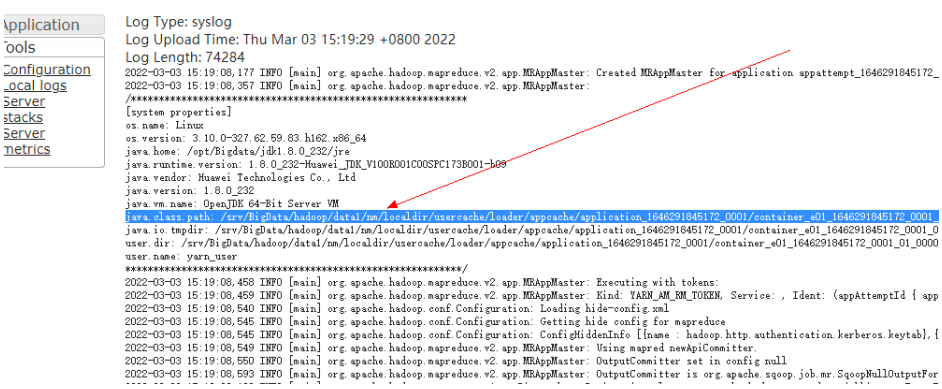
Procedure

Step 1 Use the **grep** command in the **lib** directory of Sqoop.

1. Go to the **lib** directory of Sqoop and run the **grep** command.

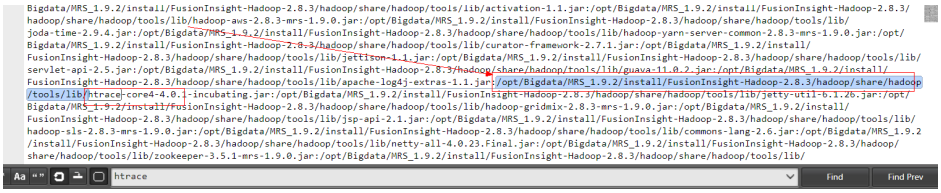


2. Go to the native Yarn page and view the error information about the running task.



Step 1 Use the **grep** command in the **lib** directory of Sqoop.

3. Copy **java.class.path** and search for **htrace-core**.



4. Copy the JAR package to the following directory:

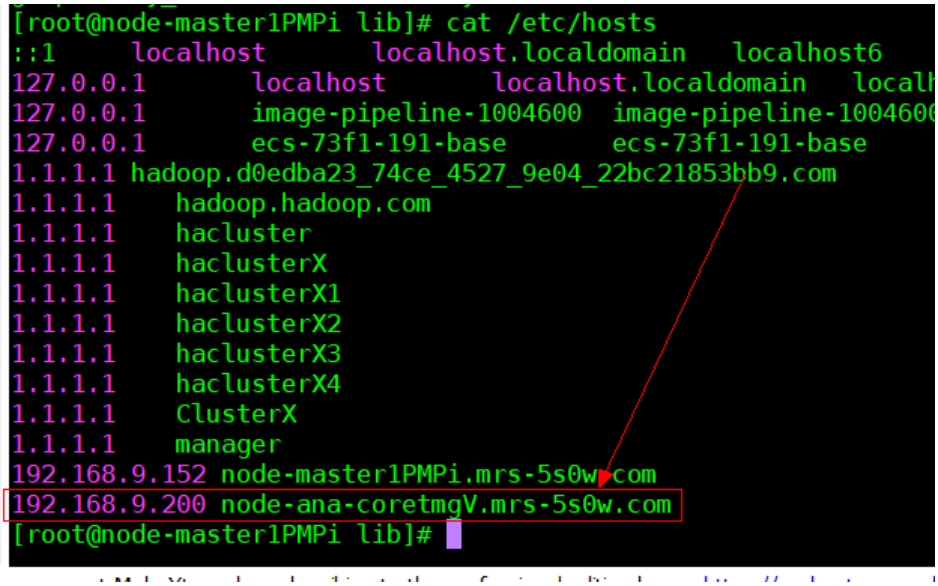
   cp /opt/Bigdata/MRS_1.9.2/install/FusionInsight-Sqoop-1.99.7/FusionInsight-Sqoop-1.99.7/server/lib/htrace-core-3.1.0-incubating.jar /opt/Bigdata/MRS_1.9.2/install/FusionInsight-Hadoop-2.8.3/hadoop/share/hadoop/common/lib/

5. Change permissions.

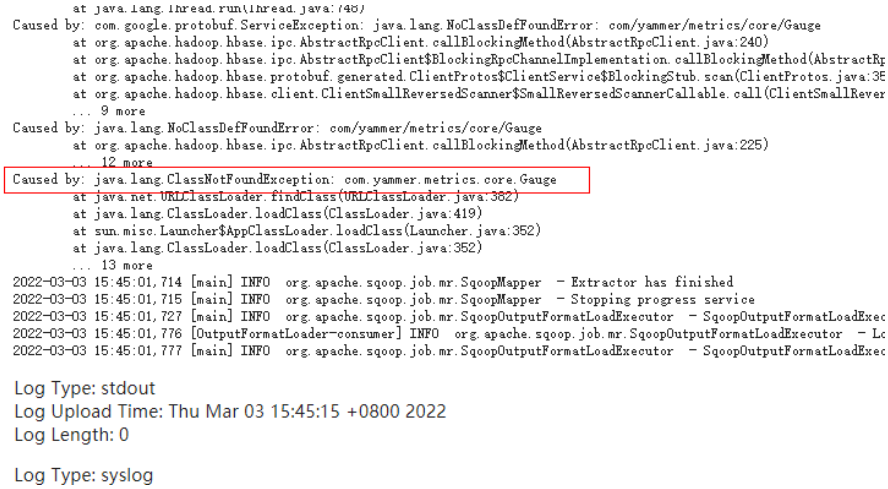
   chmod 777 *htrace-core-3.1.0-incubating.jar* (the copied JAR package)

   chown omm:ficommon *htrace-core-3.1.0-incubating.jar* (the copied JAR package)

6. View the **hosts** file and perform the same operations to copy the JAR package for all other nodes.

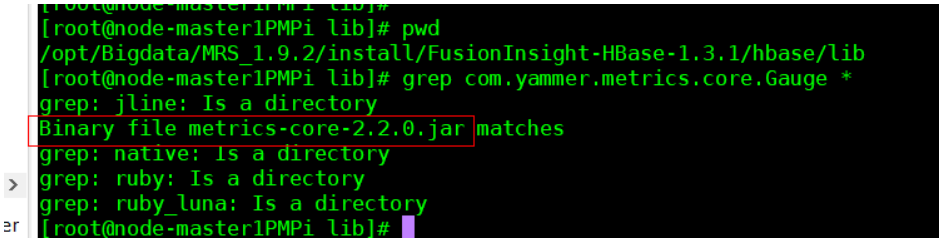


7. Run the Sqoop task again. The following error information is displayed.



Step 2 Use the **grep** command in the **lib** directory of HBase.

1. Go to the **lib** directory of HBase and run the **grep** command.



2. Copy the JAR package.

   cp /opt/Bigdata/MRS_1.9.2/install/FusionInsight-HBase-1.3.1/hbase/lib/metrics-core-2.2.0.jar /opt/Bigdata/MRS_1.9.2/install/FusionInsight-Hadoop-2.8.3/hadoop/share/hadoop/common/lib/

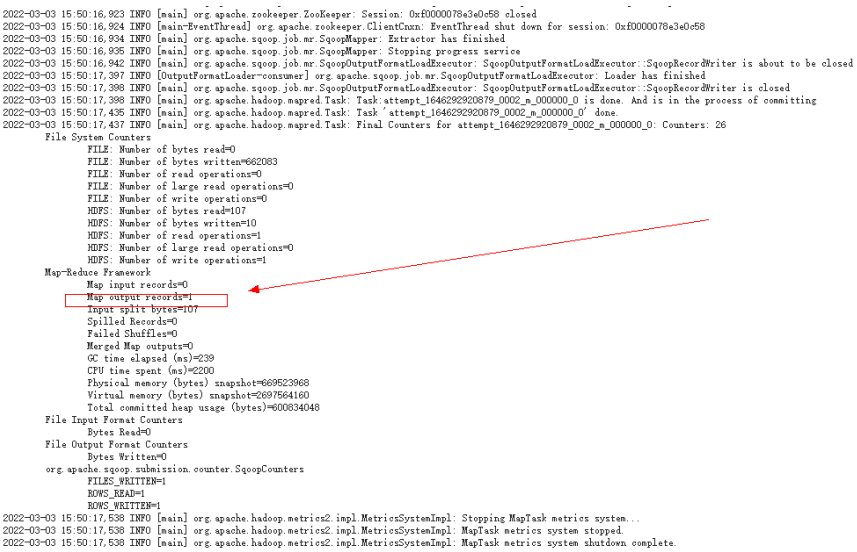
3. Change permissions.

   chmod 777 *metrics-core-2.2.0.jar* (the copied JAR package)

   chown omm:ficommon *metrics-core-2.2.0.jar* (the copied JAR package)

4. View the **hosts** file and perform the same operations to copy the JAR package for all other nodes.

5. Run the Sqoop task.



----End

Conclusion

1. Copy **htrace-core-3.1.0-incubating.jar** in the **lib** directory of Sqoop and **metrics-core-2.2.0.jar** in the **lib** directory of HBase to **/opt/Bigdata/MRS_1.9.2/install/FusionInsight-Hadoop-2.8.3/hadoop/share/hadoop/common/lib/**.
2. Change the permissions for the JAR packages to **777** and **omm:ficommon**, respectively.
3. Perform the preceding operations on all nodes and run the Sqoop task again.

16.17.4 A Format Error Is Reported When Sqoop Is Used to Export Data from Hive to MySQL 8.0

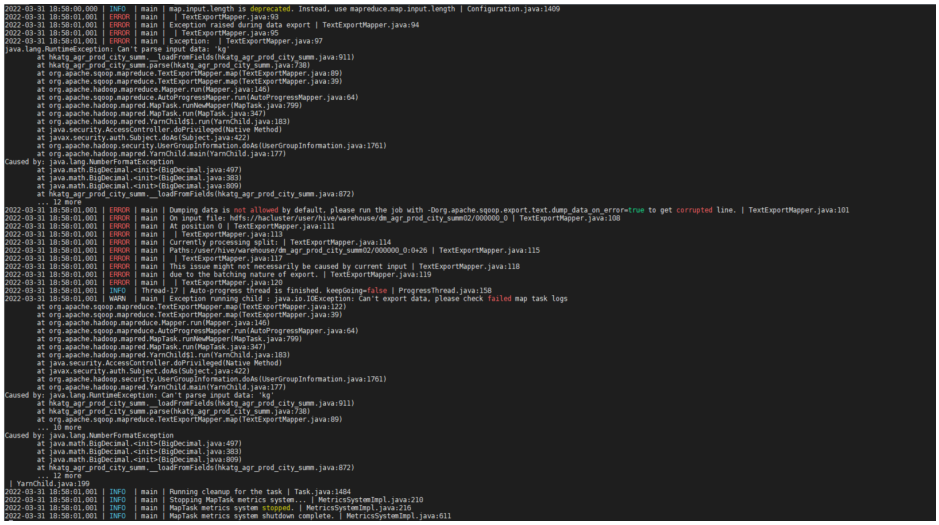
This section applies only to MRS 3.1.0 clusters.

Issue

A format error is reported when a Sqoop task is performed to export data from Hive to MySQL 8.0 in an MRS 3.1.0 cluster.

Symptom

```
2022-03-31 19:56:44,581 ERROR mapreduce.ExportJobBase: Export job failed!
2022-03-31 19:56:44,581 ERROR tool.ExportTool: Error during export:
Export job failed!
        at org.apache.sqoop.mapreduce.ExportJobBase.runExport(ExportJobBase.java:445)
        at org.apache.sqoop.manager.SqlManager.exportTable(SqlManager.java:931)
        at org.apache.sqoop.tool.ExportTool.exportTable(ExportTool.java:80)
        at org.apache.sqoop.tool.ExportTool.run(ExportTool.java:99)
        at org.apache.sqoop.Sqoop.run(Sqoop.java:147)
        at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:76)
        at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:183)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:234)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:243)
        at org.apache.sqoop.Sqoop.main(Sqoop.java:252)
```



Cause Analysis

The log shows that the format is incorrect.

Procedure

Check that the formats of delimiters and table fields are all correct. Add the **--columns** parameter to the Sqoop statement to make data formats consistent between source and target tables.

sqoop export --connect jdbc:mysql://IP address:Port number/Database name --username ***Username*** **--password** ***Password*** **--table** ***Table name*** **--columns** ***Column fields (multiple columns are separated by commas)*** **-export-dir** ***Export address*** **--fields-terminated-by** ***Delimiter*** **--input-null-string '\\N' --input-null-non-string '\\N' -m 1**

Example:

sqoop export --connect jdbc:mysql://172.16.0.6:3306/lidengpeng --username root --password Mrs@2021 --table hkatg_agr_prod_city_summ --columns year,city_name,city_code,prod_code,prod_name,prod_type,sown_area,area_unit,yield_wegt,yield_unit,total_wegt,total_wegt_unit,data_sorc_code,etl_time -export-dir hdfs://hacluster/user/hive/warehouse/dm_agr_prod_city_summ02 --fields-terminated-by ',' --input-null-string '\\N' --input-null-non-string '\\N' -m 1

16.17.5 An Error Is Reported When sqoop import Is Executed to Import PostgreSQL Data to Hive

Background

The **sqoop import** command is executed to extract data from open-source PostgreSQL to MRS HDFS or Hive.

Issue

The **sqoop** command can be executed to query the PostgreSQL database table, but an error is reported when the **sqoop import** command is executed to import data.

The authentication type 5 is not supported. Check that you have configured the pg_hba.conf file to include the client's IP address or subnet.

Cause Analysis

1. MD5 authentication for connecting to PostgreSQL fails. A whitelist needs to be configured in the **pg_hba.cnf** file.
2. When the **sqoop import** command is executed, a MapReduce job is started. The PostgreSQL driver package **gsjdbc4-*.jar** exists in the MRS Hadoop installation directory **/opt/Bigdata/FusionInsight_HD_*/1_*_DataNode/install/hadoop/share/hadoop/common/lib**, which is incompatible with the open-source PostgreSQL service. As a result, an error is reported.

Procedure

1. Configure a whitelist in the **pg_hba.cnf** file.
2. Delete the **gsjdbc4-*.jar** packages from all core nodes, and add the PostgreSQL JAR package to **sqoop/lib**.

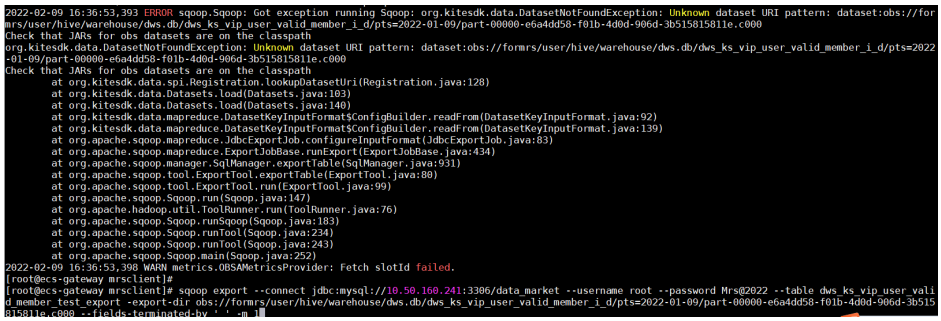
   mv /opt/Bigdata/FusionInsight_HD_*/1_*_DataNode/install/hadoop/share/hadoop/common/lib/gsjdbc4-*.jar /tmp

16.17.6 Sqoop Failed to Read Data from MySQL and Write Parquet Files to OBS

Issue

An error is reported when Sqoop reads MySQL data and writes the data to OBS in Parquet format. However, the data can be successfully written to OBS if the Parquet format is not specified.

Symptom



Cause Analysis

Parquet does not support Hive 3. Data can be written using HCatalog.

Procedure

Use HCatalog to write data: Specify the Hive database and table in parameters and modify the SQL statement in the script.

Details are as follows:

Original script:

sqoop import --connect 'jdbc:mysql://10.160.5.65/xxx_pos_online_00?zeroDateTimeBehavior=convertToNull' --username root --password Mrs@2022

--split-by id

--num-mappers 2

--query 'select * from pos_remark where 1=1 and $CONDITIONS'

--target-dir obs://za-test/dev/xxx_pos_online_00/pos_remark

--delete-target-dir

--null-string '\\N'

--null-non-string '\\N'

--as-parquetfile

Modified script:

sqoop import --connect 'jdbc:mysql://10.160.5.65/xxx_pos_online_00?zeroDateTimeBehavior=convertToNull' --username root --password Mrs@2022

--split-by id

--num-mappers 2

--query 'select id,pos_case_id,pos_transaction_id,remark,update_time,update_user,is_deleted,creator,modifier,gmt_created,gmt_modified,update_user_id,tenant_code from pos_remark where 1=1 and $CONDITIONS'

--hcatalog-database xxx_dev

--hcatalog-table ods_pos_remark

16.18 Using Storm

16.18.1 Invalid Hyperlink of Events on the Storm UI

Issue

The hyperlink of events on the Storm UI is invalid.

Symptom

After submitting a topology, a user cannot view topology data processing logs and the events hyperlink is invalid.

Cause Analysis

The function of viewing topology data processing logs is disabled by default when a topology is submitted in an MRS cluster.

Procedure

Step 1 Log in to the Storm web UI.

- For MRS 2.x and earlier versions: Choose **Storm**. On the **Storm WebUI** page, click any UI link to open the Storm web UI.

  NOTE

  When accessing the Storm web UI for the first time, you must add the address to the trusted site list.

- For MRS 3.x or later: Choose **Storm** > **Overview**. On the **Storm WebUI** in the **Basic Information** area, click any UI link to open the Storm web UI.

Step 2 In the **Topology Summary** area, click the desired topology to view details.

Step 3 In the **Topology actions** area, click **Kill** to delete the submitted Storm topology.

Step 4 Submit the Storm topology again and enable the function of viewing topology data processing logs. Add the **topology.eventlogger.executors** parameter and set it to a positive integer when submitting the Storm topology. Example:

storm jar ***Path of the topology package Class name of the topology Main method Topology name*** **-c topology.eventlogger.executors=X**

Step 5 In the **Topology Summary** area on the Storm UI, click the desired topology to view details.

Step 6 In the **Topology actions** area, click **Debug**, specify the data sampling percentage, and click **OK**.

Step 7 Click the **Spouts** or **Bolts** task name of the topology. In **Component summary**, click **events** to view data processing logs.

NOTE

To enable the function of viewing topology data processing logs of the specified **Spouts** or **Bolts** task, click the **Spouts** or **Bolts** task name of the topology, click **Debug** in the **Topology actions** area, and enter the data sampling percentage.

----End

16.18.2 Failed to Submit a Topology

Symptom

An MRS streaming cluster is installed, and ZooKeeper, Storm, as well as Kafka are installed in the cluster.

A topology fails to be submitted by running commands on the client.

Possible Causes

- The Storm service is abnormal.
- The client user is not authenticated or the authentication has expired.
- The **storm.yaml** file in the submitted topology conflicts with that on the server.

Cause Analysis

A user fails to submit the topology. The possible cause is that the client or Storm is faulty.

1. Check the Storm status.

   MRS Manager:

   Log in to MRS Manager. On the MRS Manager page, choose **Services** > **Storm** to check the status of Storm. The status is **Good**, and the monitoring metrics are correctly displayed.

   FusionInsight Manager:

   For MRS 3.*x* or later: Log in to FusionInsight Manager. Choose **Cluster** > **Services** > **Storm** to check the status of Storm. It is found that the status is **Good** and the monitoring metrics are correctly displayed.

2. Check the submission logs of the client. The logs contain "KeeperExceptionSessionExpireException".

```
org.apache.zookeeper.KeeperException$SessionExpiredException: KeeperErrorCode = Session expired
        at org.apache.zookeeper.KeeperException.create(KeeperException.java:131) ~[zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.curator.framework.imps.CuratorFrameworkImpl.checkBackgroundRetry(CuratorFrameworkImpl.java:710) [curator-framework-2.5.0.jar:na]
        at org.apache.curator.framework.imps.CuratorFrameworkImpl.processBackgroundOperation(CuratorFrameworkImpl.java:510) [curator-framework-2.5.0.jar:na]
        at org.apache.curator.framework.imps.BackgroundSyncImpl$1.processResult(BackgroundSyncImpl.java:50) [curator-framework-2.5.0.jar:na]
        at org.apache.zookeeper.ClientCnxn$EventThread.processEvent(ClientCnxn.java:484) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.zookeeper.ClientCnxn$EventThread.queuePacket(ClientCnxn.java:498) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.zookeeper.ClientCnxn.finishPacket(ClientCnxn.java:731) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.zookeeper.ClientCnxn.conLossPacket(ClientCnxn.java:765) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.zookeeper.ClientCnxn.access$2700(ClientCnxn.java:97) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.zookeeper.ClientCnxn$SendThread.cleanup(ClientCnxn.java:1391) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
        at org.apache.zookeeper.ClientCnxn$SendThread.run(ClientCnxn.java:1314) [zookeeper-3.5.0.jar:3.5.0-V100R002C00B109]
2016-05-31 09:25:24 | INFO  | [main] | Session: 0x10027394600ab4b closed | org.apache.zookeeper.ZooKeeper (ZooKeeper.java:968)
Exception in thread "main" java.lang.RuntimeException: Exception while initializing NimbusLeaderElections
        at backtype.storm.nimbus.NimbusLeaderElections.init(NimbusLeaderElections.java:86)
        at backtype.storm.utils.NimbusClient.getConfiguredClient(NimbusClient.java:39)
        at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:159)
        at backtype.storm.StormSubmitter.submitTopologyWithProgressBar(StormSubmitter.java:256)
        at backtype.storm.StormSubmitter.submitTopologyWithProgressBar(StormSubmitter.java:236)
        at storm.starter.WordCountTopology.main(WordCountTopology.java:94)
Caused by: org.apache.zookeeper.KeeperException$ConnectionLossException: KeeperErrorCode = ConnectionLoss for /storm/nimbus-leader
        at org.apache.zookeeper.KeeperException.create(KeeperException.java:99)
        at org.apache.zookeeper.KeeperException.create(KeeperException.java:51)
        at org.apache.zookeeper.ZooKeeper.exists(ZooKeeper.java:1501)
        at org.apache.curator.framework.imps.ExistsBuilderImpl$2.call(ExistsBuilderImpl.java:172)
        at org.apache.curator.framework.imps.ExistsBuilderImpl$2.call(ExistsBuilderImpl.java:161)
        at org.apache.curator.RetryLoop.callWithRetry(RetryLoop.java:107)
        at org.apache.curator.framework.imps.ExistsBuilderImpl.pathInForeground(ExistsBuilderImpl.java:157)
        at org.apache.curator.framework.imps.ExistsBuilderImpl.forPath(ExistsBuilderImpl.java:148)
        at org.apache.curator.framework.imps.ExistsBuilderImpl.forPath(ExistsBuilderImpl.java:36)
        at backtype.storm.nimbus.NimbusLeaderElections.init(NimbusLeaderElections.java:66)
        ... 5 more
```

The preceding error occurs because security authentication is not performed before the topology is submitted or the TGT expires after authentication.

For details about the solution, see **Step 1**.

3. Check the client submission log. It is found that the "ExceptionInIntializerError" exception information is printed, and the message "Found multiple storm.yaml resources" is displayed. The following is an example:

```
Exception in thread "main" java.lang.ExceptionInInitializerError
        at backtype.storm.topology.TopologyBuilder.createTopology(TopologyBuilder.java:106)
        at com.huawei.streaming.storm.example.wordcount.WordCountTopology.cmdSubmit(WordCountTopology.java:117)
        at com.huawei.streaming.storm.example.wordcount.WordCountTopology.submitTopology(WordCountTopology.java:80)
        at com.huawei.streaming.storm.example.wordcount.WordCountTopology.main(WordCountTopology.java:71)
Caused by: java.lang.RuntimeException: Found multiple storm.yaml resources. You're probably bundling the Storm jars with your topology jar.
        at backtype.storm.utils.Utils.findAndReadConfigFile(Utils.java:151)
        at backtype.storm.utils.Utils.readStormConfig(Utils.java:206)
        at backtype.storm.utils.Utils.<clinit>(Utils.java:70)
        ... 4 more
```

This error occurs because the **storm.yaml** file in the service JAR package conflicts with that on the server.

For details about the solution, see **Step 2**.

4. If the fault is not caused by the preceding reasons, see **Topology Submission Fails and the Message "Failed to check principle for keytab" Is Displayed**.

Solution

Step 1 An authentication error occurs.

1. Log in to the node where the client resides and switch to the client directory.
2. Run the following command to submit the task again: (Replace the service JAR package and topology based on the site requirements.)

   source bigdata_env

   kinit *Username*

   storm jar storm-starter-topologies-0.10.0.jar storm.starter.WordCountTopology test

Step 2 The topology package is abnormal.

Check the service JAR package, delete the **storm.yaml** file from the service JAR package, and submit the task again.

----End

16.18.3 Topology Submission Fails and the Message "Failed to check principle for keytab" Is Displayed

Symptom

An MRS streaming cluster in security mode is installed, and ZooKeeper, Storm, and Kafka are installed in the cluster.

When a topology is defined to access components such as HDFS and HBase and the topology fails to be submitted using client commands.

Possible Causes

- The submitted topology does not contain the keytab file of the user.
- The keytab file contained in the submitted topology is inconsistent with the user who submits the topology.
- The **user.keytab** file exists in the **/tmp** directory on the client, and the owner is not the running user.

Cause Analysis

1. Check the logs. Error information "Can not found user.keytab in storm.jar" is found. Details are as follows:

   ```
   [main] INFO  b.s.StormSubmitter - Get principle for stream@HADOOP.COM success
   [main] ERROR b.s.StormSubmitter - Can not found user.keytab in storm.jar.
   Exception in thread "main" java.lang.RuntimeException: Failed to check principle for keytab
   at backtype.storm.StormSubmitter.submitTopologyAs(StormSubmitter.java:219)
   at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:292)
   at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:176)
   at com.xxx.streaming.storm.example.hbase.SimpleHBaseTopology.main(SimpleHBaseTopology.java:77)
   ```

   Check the JAR file of the submitted topology. It is found that the keytab file is not contained.

2. Check the logs. Error information "The submit user is invalid,the principle is" is found. Details are as follows:

   ```
   [main] INFO  b.s.StormSubmitter - Get principle for stream@HADOOP.COM success
   [main] WARN  b.s.s.a.k.ClientCallbackHandler - Could not login: the client is being asked for a password, but the  client code does not currently support obtaining a password from the user. Make sure that the client is configured to use a ticket cache (using the JAAS configuration setting 'useTicketCache=true)' and restart the client. If you still get this message after that, the TGT in the ticket cache has expired and must be manually refreshed. To do so, first determine if you are using a password or a keytab. If the former, run kinit in a Unix shell in the environment of the user who is running this client using the command 'kinit <princ>' (where <princ> is the name of the client's Kerberos principal). If the latter, do 'kinit -k -t <keytab> <princ>' (where <princ> is the name of the Kerberos principal, and <keytab> is the location of the keytab file). After manually refreshing your cache, restart this client. If you continue to see this message after manually refreshing your cache, ensure that your KDC host's clock is in sync with this host's clock.
   [main] ERROR b.s.StormSubmitter - The submit user is invalid,the principle is : stream@HADOOP.COM
   Exception in thread "main" java.lang.RuntimeException: Failed to check principle for keytab
   at backtype.storm.StormSubmitter.submitTopologyAs(StormSubmitter.java:219)
   at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:292)
   at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:176)
   at com.xxx.streaming.storm.example.hbase.SimpleHBaseTopology.main(SimpleHBaseTopology.java:77)
   ```

   The authenticated user used to submit the topology is **stream**. However, the system displays a message indicating that the submit user is invalid during topology submission, indicating that the internal verification fails.

3. Check the JAR file of the submitted topology. It is found that the keytab file is contained.

The principal parameter is set to **zmk_kafka** in the **user.keytab** file.

```
[root@8-5-148-6 client]# klist -kt user.keytab
Keytab name: FILE:user.keytab
KVNO Timestamp         Principal
---- ----------------- --------------------------------------------------------
   1 12/19/16 16:28:17 zmk_kafka@HADOOP.COM
   1 12/19/16 16:28:17 zmk_kafka@HADOOP.COM
```

It is found that the authenticated user does not match the principal in the **user.keytab** file.

4. Check the logs and find the error information "Delete the tmp keytab file failed, the keytab file is:/tmp/user.keytab". The detailed information is as follows:

```
[main] WARN b.s.StormSubmitter - Delete the tmp keytab file failed, the keytab file is : /tmp/
user.keytab
[main] ERROR b.s.StormSubmitter - The submit user is invalid,the principle is : hbase1@HADOOP.COM
Exception in thread "main" java.lang.RuntimeException: Failed to check principle for keytab
at backtype.storm.StormSubmitter.submitTopologyAs(StormSubmitter.java:213)
at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:286)
at backtype.storm.StormSubmitter.submitTopology(StormSubmitter.java:170)
at com.touchstone.storm.cmcc.CmccDataHbaseTopology.main(CmccDataHbaseTopology.java:183)
```

Check the **/tmp** directory. It is found that the **user.keytab** file exists and the file owner is not the running user.

Solution

- Ensure that the **user.keytab** file is carried when the topology is submitted.
- Ensure that the user for submitting the topology is the same as that of the **user.keytab** file.
- Delete the **user.keytab** file from the **/tmp** directory.

16.18.4 The Worker Log Is Empty After a Topology Is Submitted

Symptom

After a topology is remotely submitted in Eclipse, the detailed information about the topology cannot be viewed on the Storm web UI, and the Worker node where Bolt and Spout of each topology are located keeps changing. The Worker log is empty.

Possible Causes

The Worker process fails to be started, triggering Nimbus to re-allocate tasks and start the Worker process on other Supervisors. The Worker process continues to restart. As a result, the Worker node keeps changing, and the Worker log is empty. The possible causes of the Worker process startup failure are as follows:

- The submitted JAR package contains the **storm.yaml** file.

  Storm specifies that each classpath can contain only one **storm.yaml** file. If there is more than one **storm.yaml** file, an exception occurs. Use the Storm client to submit the topology. The classpath configuration of the client is different from the classpath configuration of Eclipse. The client automatically loads the JAR package of the user to classpath. As a result, two **storm.yaml** files exist in classpath.

- The initialization of the Worker process takes a long time, which exceeds the Worker startup timeout period set in the Storm cluster. As a result, the Worker process is killed and reallocated.

Troubleshooting Process

1. Use the Storm client to submit the topology and check whether the **storm.yaml** file is duplicate.
2. Repack the JAR file and submit the topology again.
3. Modify the Worker startup timeout parameter in the Storm cluster.

Procedure

Step 1 If the Worker log is empty after the topology is remotely submitted using Eclipse, use the Storm client to submit the JAR package corresponding to the topology and view the prompt message.

For example, if the JAR package contains two **storm.yaml** files in different paths, the following information is displayed:

```
Exception in thread "main" java.lang.ExceptionInInitializerError
  at com.xxx.streaming.storm.example.WordCountTopology.createConf(WordCountTopology.java:132)
  at com.xxx.streaming.storm.example.WordCountTopology.remoteSubmit(WordCountTopology.java:120)
  at com.xxx.streaming.storm.example.WordCountTopology.main(WordCountTopology.java:101)
 Caused by: java.lang.RuntimeException: Found multiple storm.yaml resources. You're probably bundling the Storm jars with your topology jar. [jar:file:/opt/xxx/fi_client/Streaming/streaming-0.9.2/bin/stormDemo.jar!/storm.yaml, file:/opt/xxx/fi_client/Streaming/streaming-0.9.2/conf/storm.yaml]
  at backtype.storm.utils.Utils.findAndReadConfigFile(Utils.java:151)
  at backtype.storm.utils.Utils.readStormConfig(Utils.java:206)
  at backtype.storm.utils.Utils.<(Utils.java:70)>
```

Step 2 Compress the JAR package again. Ensure that the package does not contain the **storm.yaml** file and JAR packages related to **log4j** and **slf4j-log4j**.

Step 3 Use IntelliJ IDEA to remotely submit the new JAR package.

Step 4 Check whether the topology details and Worker logs can be viewed on the web UI.

Step 5 On MRS Manager, modify the Worker startup timeout parameter of the Storm cluster (for details about the parameter description, see **Related Information**). Save the modification, and restart the Storm service.

- MRS Manager: Log in to MRS Manager and choose **Services** > **Storm** > **Configuration**.
- FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Storm** > **Configuration**.

Step 6 Submit the JAR package to be run again.

----End

Related Information

1. The **nimbus.task.launch.secs** and **supervisor.worker.start.timeout.secs** parameters indicate the topology startup timeout tolerance of the Nimbus and supervisor, respectively. Generally, the value of **nimbus.task.launch.secs** must be greater than or equal to that of **supervisor.worker.start.timeout.secs**. It is recommended that the value of

nimbus.task.launch.secs be slightly greater or equal to that of **supervisor.worker.start.timeout.secs**. Otherwise, the task reallocation efficiency will be affected.

- **nimbus.task.launch.secs**: If the Nimbus does not receive the heartbeat message sent by the topology task within the period specified by this parameter, the Nimbus re-allocates the topology to another supervisor and updates the task information in ZooKeeper. The supervisor reads the task information in ZooKeeper and compares it with the topology started. If the topology does not belong to the supervisor, the supervisor deletes the metadata of the topology, that is, the **/srv/Bigdata/streaming_data/stormdir/supervisor/stormdist/{worker-id}** directory.
- **supervisor.worker.start.timeout.secs**: After the supervisor starts a worker, if no heartbeat message is received from the worker within the period specified by this parameter, the supervisor stops the worker and waits for worker rescheduling. Generally, the value of this parameter is increased when the service startup takes a long time to ensure that the worker can be started successfully.

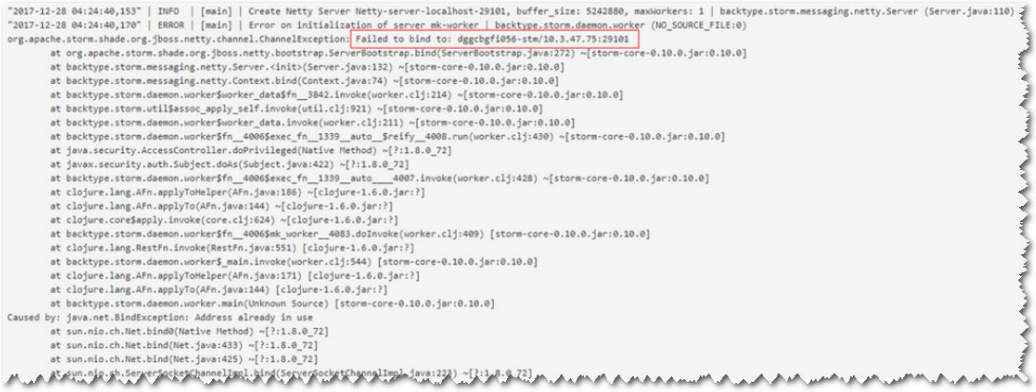
  If the value of **supervisor.worker.start.timeout.secs** is greater than that of **nimbus.task.launch.secs**, the worker is still started before the tolerance time of supervisor ends. However, the Nimbus considers that the service startup times out and allocates the service to another host. The background thread of the supervisor finds that the tasks are inconsistent and deletes the metadata of the topology. As a result, when the worker attempts to read **stormconf.ser** during startup, the file does not exist, and "FileNotFoundException" is thrown.

2. The **nimbus.task.timeout.secs** and **supervisor.worker.timeout.secs** parameters indicate the timeout tolerance time for the Nimbus and supervisor to report heartbeat messages during topology running. Generally, the value of **nimbus.task.timeout.secs** must be slightly greater than or equal to that of **supervisor.worker.timeout.secs**.

16.18.5 Worker Runs Abnormally After a Topology Is Submitted and Error "Failed to bind to:host:ip" Is Displayed

Symptom

After the service topology is submitted, the Worker cannot be started normally. Check the Worker log. The log records "Failed to bind to: host:ip."



Possible Causes

The random port range is incorrectly configured.

Troubleshooting Process

1. Check related information in the Worker log.
2. Check the process information about the bond port.
3. Check the random port range.

Cause Analysis

1. Use SSH to log in to the host where the Worker fails to be started and run the **netstat -anp | grep** <*port*> command to check the ID of the process that occupies the port. In the preceding command, change *port* to the actual port number.
2. Run the **ps -ef | grep** <*pid*> command to view process details. In the command, *pid* indicates the actual process ID.

It is found that the worker process occupies the port. This process is another topology service process. According to the process details, port 29122 is allocated to the process.

3. Run the **lsof -i:**<*port*> command to view connection details. In the preceding command, change *port* to the actual port number.

It is found that port 29101 connects to port 21005 of the peer end, and port 21005 is the Kafka server port.

It indicates that the service layer connects to Kafka to obtain messages as a client. Service ports are allocated based on the random port range of the OS.

4. Run the **cat /proc/sys/net/ipv4/ip_local_port_range** command to check the random port range.

5. It is found that the random port range is too large and conflicts with the service port range of MRS.

NOTE

The MRS service port number ranges from 20000 to 30000.

Procedure

Step 1 Modify the random port range.

```
vi /proc/sys/net/ipv4/ip_local_port_range
32768 61000
```

Step 2 Stop the service process that occupies the service port to release the port. (Stop the service topology.)

----End

16.18.6 "well-known file is not secure" Is Displayed When the jstack Command Is Used to Check the Process Stack

Symptom

Run the **jstack** command to check the process stack information. The error message "well-known file is not secure" is displayed.

```
omm@hadoop02:~> jstack 62517
62517: well-known file is not secure
```

Cause Analysis

1. The user running the **jstack** command is inconsistent with the user submitting the process for viewing the pid information.
2. Storm uses the feature of differentiating users for implementing tasks. When the worker process is started, the process UID and GID are changed to the user submitting the task and ficommon. This way, logviewer can access logs of the worker process and only log file permission 640 is open. After the user is changed, the **jstack** and **jmap** commands fail to be executed for the worker process, because the default GID of the user is not ficommon. You need to run the ldap command to change the user GID to 9998 (ficommon).

Solution

You can use either of the following two methods to resolve the problem:

Method 1: View the process stack on the native Storm page.

Step 1 Log in to the native Storm page.

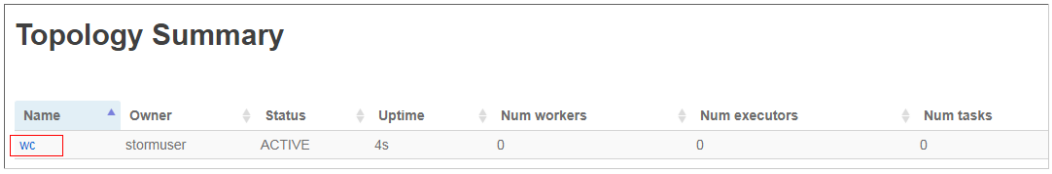
MRS Manager:

1. Access MRS Manager.
2. Choose **Services** > **Storm**. In **Storm WebUI** of **Storm Summary**, click any UI link to access the Storm WebUI.

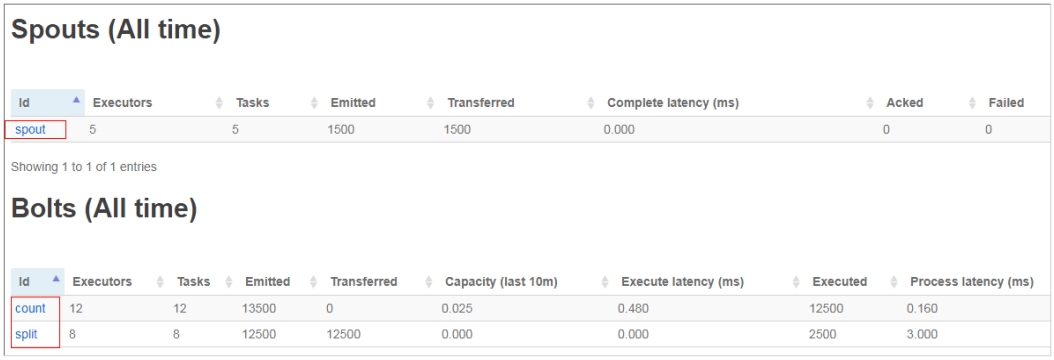
FusionInsight Manager:

1. Log in to FusionInsight Manager.
2. On Manager, choose **Cluster > Service** > **Storm**. On the **Storm WebUI** page of **Overview**, click any UI link to open the Storm WebUI.

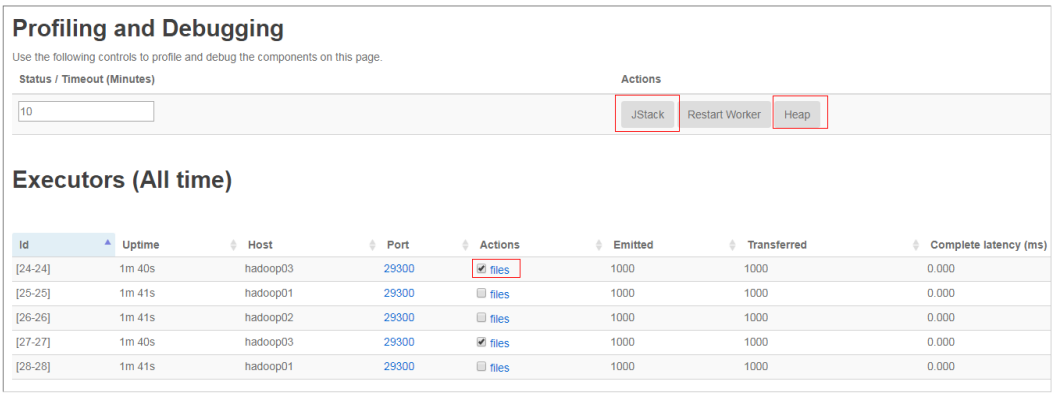
Step 2 Select the topology to be viewed.



Step 3 Select the spout or bolt to be viewed.



Step 4 Select the log file of the node to be viewed, and then click **JStack** or **Heap**. **JStack** corresponds to the stack information, and **Heap** corresponds to the heap information.



----End

Method 2: View the process stack by modifying user-defined parameters.

Step 1 Access the Storm parameter configuration page.

MRS Manager: Log in to MRS Manager, choose **Services** > **Storm** > **Service Configuration**, and select **All** from the **Type** drop-down list.

Operation on FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Yarn** > **Configurations** > **All Configurations**.

Step 2 In the navigation tree on the left, choose **supervisor** > **Customize** and add the variable **supervisor.run.worker.as.user=false**.

Step 3 Click **Save Configuration** and select **Restart the affected services or instances**. Click **OK** to restart the services.

Step 4 Submit the topology again.

Step 5 Switch to the **omm** user on the background node and run the **jps** command to view the PID of the worker process.

```
omm@hadoop02:~> jps | grep worker
22485 worker
111402 worker
```

Step 6 Run the **jstack pid** command to view the jstack information.

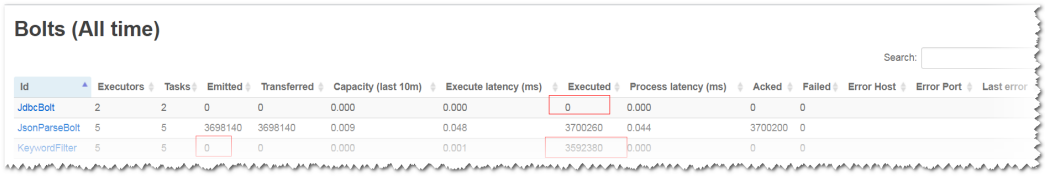
```
omm@hadoop02:~> jstack 22485
2018-05-26 08:46:24
Full thread dump Java HotSpot(TM) 64-Bit Server VM (25.144-b01 mixed mode):

"Attach Listener" #82 daemon prio=9 os_prio=0 tid=0x0000000001c95000 nid=0xb840 waiting on condition [0x0000000000000000]
   java.lang.Thread.State: RUNNABLE

"pool-14-thread-1" #81 daemon prio=5 os_prio=0 tid=0x00007f7ebc931000 nid=0x6113 waiting on condition [0x00007f7eb5ddf000]
   java.lang.Thread.State: TIMED_WAITING (parking)
        at sun.misc.Unsafe.park(Native Method)
        - parking to wait for  <0x00000000dfe820a0> (a java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
        at java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:215)
        at java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.awaitNanos(AbstractQueuedSynchronizer.java:2078)
        at java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.take(ScheduledThreadPoolExecutor.java:1093)
        at java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.take(ScheduledThreadPoolExecutor.java:809)
        at java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1074)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1134)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:748)
```

----End

16.18.7 When the Storm-JDBC plug-in is used to develop Oracle write Bolts, data cannot be written into the Bolts.

Symptom

When the Storm-JDBC plug-in is used to develop Oracle write Bolts, the Oracle database can be connected, but data cannot be written to the Oracle database.

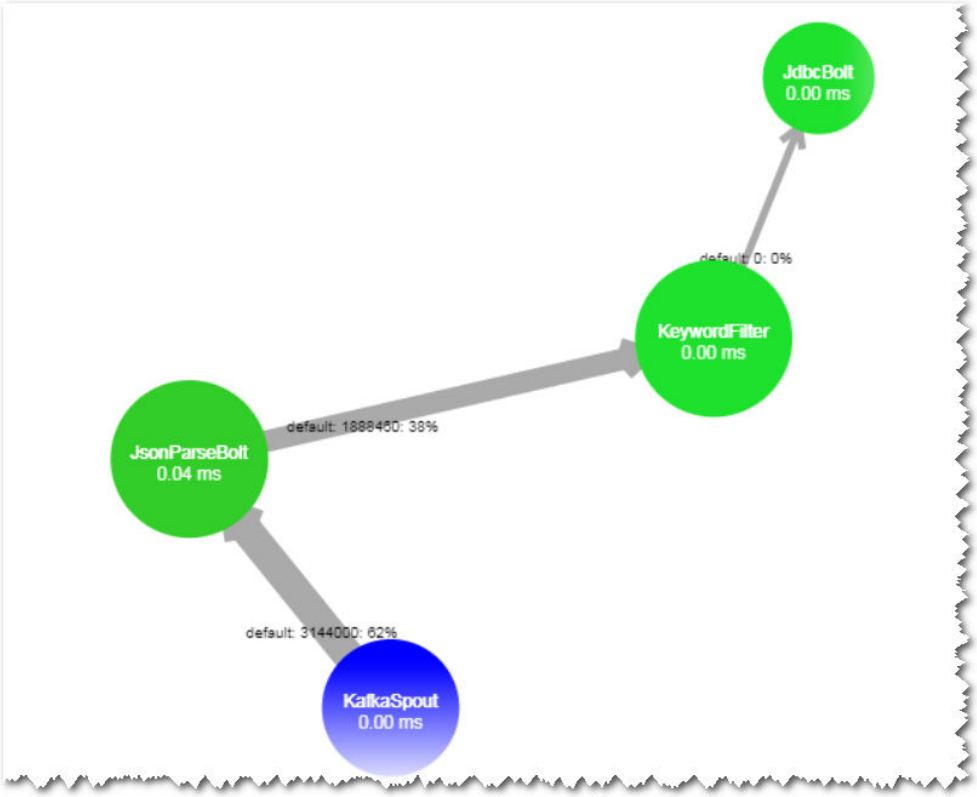


Possible Causes

- The topology definition is incorrect.
- The definition of the database table result is incorrect.

Cause Analysis

1. On the Storm web UI, check the DAG of the topology. The DAG is consistent with the topology definition.

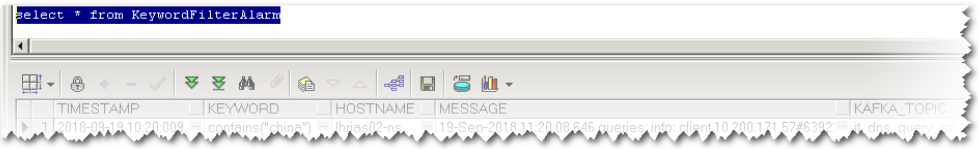


2. The definition of the KeyWordFilter Bolt is consistent with the **expParser** field.

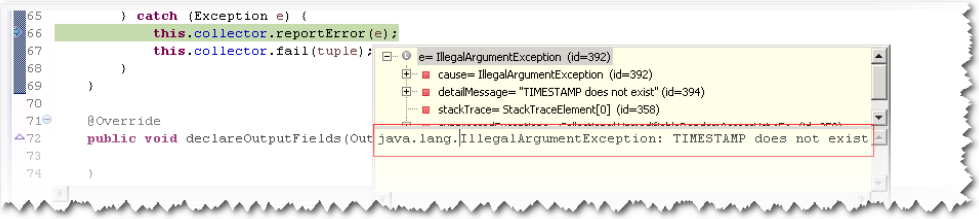
```
@Override
public void declareOutputFields(OutputFieldsDeclarer declarer)
{
    declarer.declare(new Fields("timestamp","keyword","hostname","message","kafka_topic") );
```

```
    if( flag )
    {
        String keyword = expParser.getKeyword();
        System.out.println( message );
        collector.emit(new Values( timestamp, keyword , hostname , message, kafka_topic ));
```

3. View the table definition in the Oracle database. The field name is in uppercase, which is inconsistent with flow definition field name.



4. When the execute method is debugged independently, it is found that the thrown field does not exist.



Procedure

The field name of the stream definition is changed to uppercase letters, which is the same as that defined in the database table.

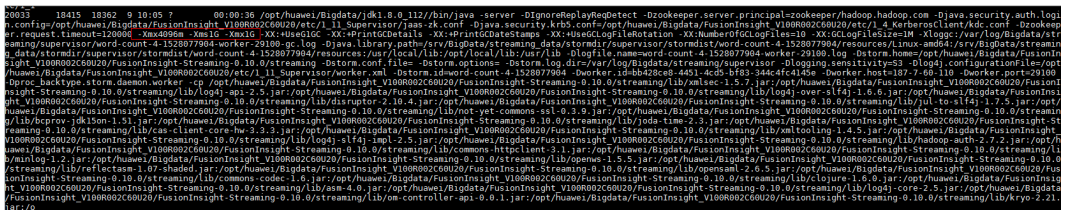
16.18.8 The GC Parameter Configured for the Service Topology Does Not Take Effect

Symptom

The **topology.worker.childopts** parameter in the service topology code does not take effect. The key log is as follows:

```
[main] INFO b.s.StormSubmitter - Uploading topology jar /opt/jar/example.jar to assigned location: /srv/
BigData/streaming/stormdir/nimbus/inbox/stormjar-8d3b778d-69ea-4fbe-ba88-01aa2036d753.jar
Start uploading file '/opt/jar/example.jar' to '/srv/BigData/streaming/stormdir/nimbus/inbox/
stormjar-8d3b778d-69ea-4fbe-ba88-01aa2036d753.jar' (65574612 bytes)
[==================================================] 65574612 / 65574612
File '/opt/jar/example.jar' uploaded to '/srv/BigData/streaming/stormdir/nimbus/inbox/
stormjar-8d3b778d-69ea-4fbe-ba88-01aa2036d753.jar' (65574612 bytes)
[main] INFO b.s.StormSubmitter - Successfully uploaded topology jar to assigned location: /srv/BigData/
streaming/stormdir/nimbus/inbox/stormjar-8d3b778d-69ea-4fbe-ba88-01aa2036d753.jar
[main] INFO b.s.StormSubmitter - Submitting topology word-count in distributed mode with conf
{"topology.worker.childopts":"-
Xmx4096m","storm.zookeeper.topology.auth.scheme":"digest","storm.zookeeper.topology.auth.payload":"-59
15065013522446406:-6421330379815193999","topology.workers":1}
[main] INFO b.s.StormSubmitter - Finished submitting topology: word-count
```

The following worker process information is displayed after the **ps -ef | grep worker** command is executed:



Cause Analysis

1. **topology.worker.gc.childopts**, **topology.worker.childopts**, and **worker.gc.childopts** (server parameters) have priorities: **topology.worker.gc.childopts** > **worker.gc.childopts** > **topology.worker.childopts**.
2. If the client parameter **topology.worker.childopts** is set, this parameter and the server parameter **worker.gc.childopts** are configured together. However, for two same parameters, one of them will be overwritten by the other parameter after it. Take parameter **-Xmx**, as shown in the red box of the preceding figure, as an example, parameter **-Xmx1G** overwrites **-Xmx4096m**.
3. If parameter **topology.worker.gc.childopts** is configured on the client, the parameter **worker.gc.childopts** on the server will be replaced.

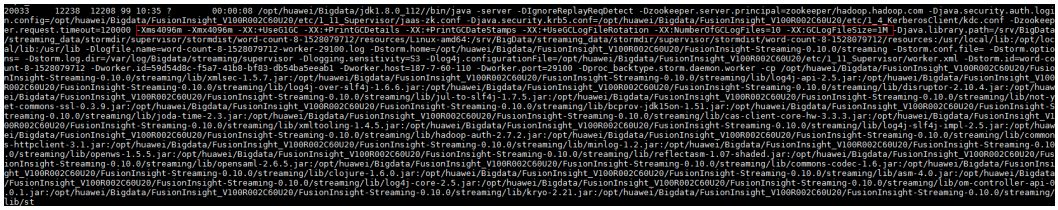
Solution

Step 1 If you want to modify the JVM parameter of the topology, you can directly modify the **topology.worker.gc.childopts** parameter in the command or modify the

parameter on the server. When **topology.worker.gc.childopts** is set to **-Xms4096m -Xmx4096m -XX:+UseG1GC -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M**:

```
[main-SendThread(10.7.61.88:2181)] INFO o.a.s.s.o.a.z.ClientCnxn - Socket connection established, initiating session, client: /10.7.61.88:44694, server: 10.7.61.88/10.7.61.88:2181
[main-SendThread(10.7.61.88:2181)] INFO o.a.s.s.o.a.z.ClientCnxn - Session establishment complete on server 10.7.61.88/10.7.61.88:2181, sessionid = 0x16037a6e5f092575, negotiated timeout = 40000
[main-EventThread] INFO o.a.s.s.o.a.c.f.s.ConnectionStateManager - State change: CONNECTED
[main] INFO b.s.u.StormBoundedExponentialBackoffRetry - The baseSleepTimeMs [1000] the maxSleepTimeMs [1000] the maxRetries [1]
[main] INFO o.a.s.s.o.a.z.Login - successfully logged in.
[main-EventThread] INFO o.a.s.s.o.a.z.ClientCnxn - EventThread shut down for session: 0x16037a6e5f092575
[main] INFO o.a.s.s.o.a.z.ZooKeeper - Session: 0x16037a6e5f092575 closed
[main] INFO b.s.StormSubmitter - Uploading topology jar /opt/jar/example.jar to assigned location: /srv/BigData/streaming/stormdir/nimbus/inbox/stormjar-86855b6b-133e-478d-b415-fa96e63e553f.jar
Start uploading file '/opt/jar/example.jar' to '/srv/BigData/streaming/stormdir/nimbus/inbox/stormjar-86855b6b-133e-478d-b415-fa96e63e553f.jar' (74143745 bytes)
[==================================================] 74143745 / 74143745
File '/opt/jar/example.jar' uploaded to '/srv/BigData/streaming/stormdir/nimbus/inbox/stormjar-86855b6b-133e-478d-b415-fa96e63e553f.jar' (74143745 bytes)
[main] INFO b.s.StormSubmitter - Successfully uploaded topology jar to assigned location: /srv/BigData/streaming/stormdir/nimbus/inbox/stormjar-86855b6b-133e-478d-b415-fa96e63e553f.jar
[main] INFO b.s.StormSubmitter - Submitting topology word-count in distributed mode with conf {"storm.zookeeper.topology.auth.scheme":"digest","storm.zookeeper.topology.auth.payload":"-7360002804241426074:-6868950379453400421","topology.worker.gc.childopts":"-Xms4096m -Xmx4096m -XX:+UseG1GC -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=10 -XX:GCLogFileSize=1M","topology.workers":1}
[main] INFO b.s.StormSubmitter - Finished submitting topology: word-count
```

Step 2 Run the **ps -ef | grep worker** command to view the worker process information:



----End

16.18.9 Internal Server Error Is Displayed When the User Queries Information on the UI

Symptom

An MRS cluster is installed, and ZooKeeper and Storm are installed in the cluster.

"Internal Server Error" is displayed when a user accesses information from the **Storm Status** page of MRS Manager.

The detailed information is as follows:

```
Internal Server Error
org.apache.thrift7.transport.TTransportException: Frame size (306030) larger than max length (1048576)!
```

Possible Causes

- Nimbus of Storm is abnormal.
- Storm cluster information exceeds the default Thrift transmission size.

Cause Analysis

1. Check the Storm service status and monitoring metrics:
   - MRS Manager: Log in to MRS Manager and choose **Services** > **Storm**. Check the Storm status. The status is **Good**, and the monitoring metrics are correctly displayed.
   - FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > *Name of the target cluster* > **Service** > **Storm**.

     Check the Storm status. It is found that the status is good and the monitoring metrics are correctly displayed.
2. Click the **Instance** tab and check the status of the Nimbus instance. The status is normal.
3. Check the Thrift configuration of the Storm cluster. It is found that **nimbus.thrift.max_buffer_size** is set to **1048576** (1 MB).
4. The preceding configuration is the same as that in the exception information, indicating that the buffer size of Thrift is less than that required by the cluster information.

Procedure

Adjust the Thrift buffer size of the Storm cluster.

Step 1 Access the Storm parameter configuration page.

- MRS Manager: Log in to MRS Manager, choose **Services** > **Storm** > **Service Configuration**, and select **All** from the **Type** drop-down list.
- Operation on FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Yarn** > **Configurations** > **All Configurations**.

Step 2 Change the value of **nimbus.thrift.max_buffer_size** to **10485760** (10 MB).

Step 3 Click Save Configuration and select **Restart the affected services or instances**. Click **OK** to restart the services.

----End

16.19 Using Ranger

16.19.1 After Ranger Authentication Is Enabled for Hive, Unauthorized Tables and Databases Can Be Viewed on the Hue Page

Issue

Although Ranger authentication is enabled for Hive, unauthorized tables and databases can be still viewed on the Hue page.

Symptom

In a normal cluster with Kerberos authentication disabled, after Ranger authentication is enabled for Hive, unauthorized tables and databases can be viewed on the Hue page.

Cause Analysis

After Ranger authentication is enabled for Hive, the default Hive policies contain two public group policies about databases. All users belong to the public group. By default, the public group is granted the permission to create tables in the default database and create other databases. Therefore, all users have the **show databases** and **show tables** permissions by default. If some users do not need to have these two permissions, you can delete the default public group policies on the Ranger web UI and grant the required user permissions.

Procedure

Step 1 Log in to the Ranger web UI.

Step 2 In the **Service Manager** area, click the Hive component name to access the Hive security access policy page.

Step 3 Click in the rows containing the **all - database** and **default database tables columns** policies.

Step 4 Delete the public group policies.

Figure 16-59 all - database policy

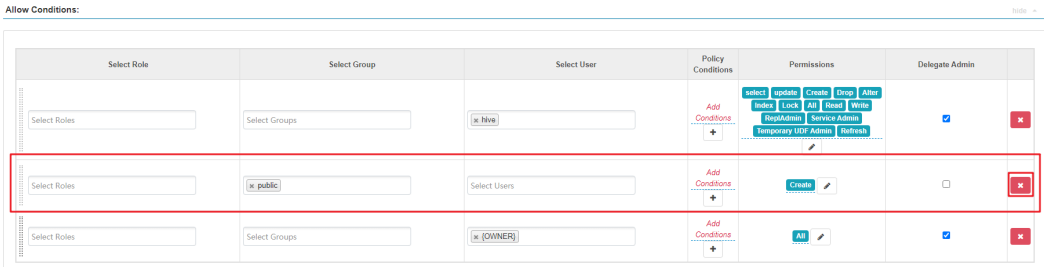
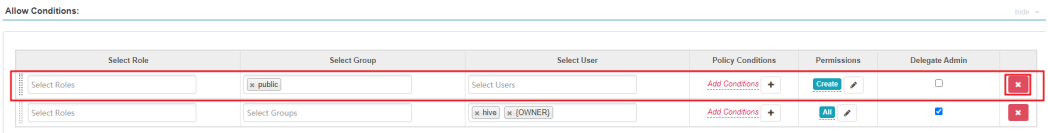


Figure 16-60 default database tables columns policy



Step 5 On the Hive security access policy page, click **Add New Policy** to add resource access policies for related users or user groups.

----End

16.20 Using Yarn

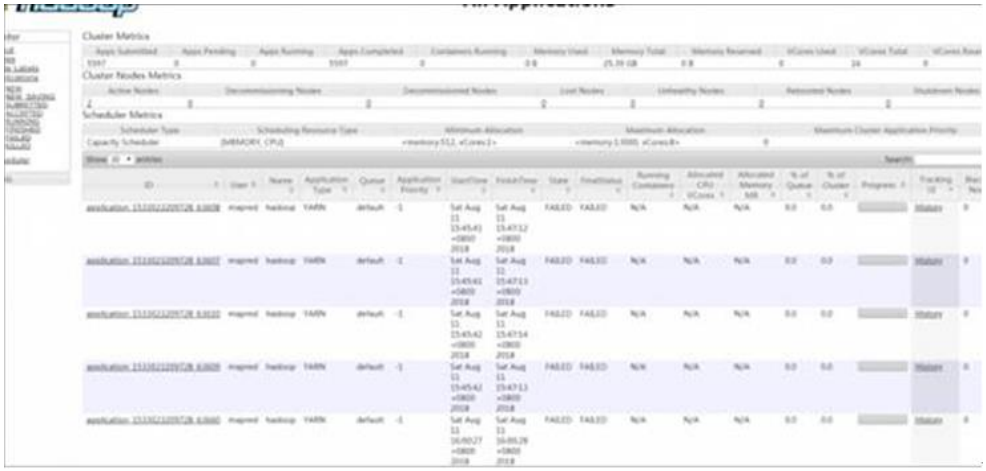
16.20.1 Plenty of Jobs Are Found After Yarn Is Started

Issue

After Yarn starts in an MRS cluster (MRS 2.x or earlier), plenty of jobs occupying resources are found.

Symptom

After the customer creates an MRS cluster and starts Yarn, plenty of jobs occupying resources are found.



Cause Analysis

- It is suspected that there are hacker attacks.
- Set the Any protocol in the inbound direction of the SG to the 0.0.0.0/0.

IPv4	Any	Any	0.0.0.0/0
IPv4	Any	Any	0.0.0.0/0
IPv4	Any	Any	0.0.0.0/0

Procedure

Step 1 Log in to the MRS management console. On the **Active Clusters** page, click the cluster name. The cluster details page is displayed.

Step 2 Click **Manage** next to **Access Manager**. The **Access MRS Manager** page is displayed.

Step 3 Click **Manage Security Group Rule** to check the security group rule configuration.

Step 4 Check whether the source address of the Any protocol in the inbound direction is 0.0.0.0/0.

Step 5 If it is 0.0.0.0/0, change the remote end of the Any protocol in the inbound direction to a specified IP address. If it is not 0.0.0.0/0, there is no need to change the value.

Step 6 After the value is changed successfully, restart the cluster VM.

----End

Summary and Suggestions

Disable the Any protocol in the inbound direction, or specify the remote end of the Any protocol in the inbound direction as the specified IP address.

Related Information

For details, see .

16.20.2 "GC overhead" Is Displayed on the Client When Tasks Are Submitted Using the Hadoop Jar Command

Symptom

When a user submits a task on the client, the client returns a memory overflow error.

```
main path:hdfs://hacluster/user/wangyou
17/09/18 08:29:57 INFO hdfs.DFSClient: Created HDFS_DELEGATION_TOKEN token 22890097 for wangyou on ha-hdfs:hacluster
17/09/18 08:29:57 INFO security.TokenCache: Got dt for hdfs://hacluster; kind: HDFS_DELEGATION_TOKEN, Service: ha-hdfs:hacluster, Ident: (HDFS_DELEGATION_TOKEN token 2
2890097 for wangyou)
17/09/18 08:29:57 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with
ToolRunner to remedy this.
17/09/18 08:32:42 INFO retry.RetryInvocationHandler: Exception while invoking getListing of class ClientNamenodeProtocolTranslatorPB over fi1-cn-003/10.113.246.10:2500
0. Trying to fail over immediately.
java.io.IOException: com.google.protobuf.ServiceException: java.lang.OutOfMemoryError: GC overhead limit exceeded
        at org.apache.hadoop.ipc.ProtobufHelper.getRemoteException(ProtobufHelper.java:47)
        at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.getListing(ClientNamenodeProtocolTranslatorPB.java:578)
        at sun.reflect.GeneratedMethodAccessor2.invoke(Unknown Source)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:497)
        at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:191)
        at org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:102)
        at com.sun.proxy.$Proxy10.getListing(Unknown Source)
        at org.apache.hadoop.hdfs.DFSClient.listPaths(DFSClient.java:1757)
        at org.apache.hadoop.hdfs.DistributedFileSystem$DirListingIterator.hasNextNoFilter(DistributedFileSystem.java:1024)
        at org.apache.hadoop.hdfs.DistributedFileSystem$DirListingIterator.hasNext(DistributedFileSystem.java:999)
        at org.apache.hadoop.mapreduce.lib.input.FileInputFormat.singleThreadedListStatus(FileInputFormat.java:304)
        at org.apache.hadoop.mapreduce.lib.input.FileInputFormat.listStatus(FileInputFormat.java:265)
        at org.apache.hadoop.mapreduce.lib.input.CombineFileInputFormat.getSplits(CombineFileInputFormat.java:217)
        at org.apache.hadoop.mapreduce.lib.input.DelegatingInputFormat.getSplits(DelegatingInputFormat.java:115)
        at org.apache.hadoop.mapreduce.JobSubmitter.writeNewSplits(JobSubmitter.java:306)
        at org.apache.hadoop.mapreduce.JobSubmitter.writeSplits(JobSubmitter.java:323)
        at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:200)
        at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1290)
        at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1287)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1673)
        at org.apache.hadoop.mapreduce.Job.submit(Job.java:1287)
```

Cause Analysis

According to the error stack, the memory overflows when the HDFS files are read during task submission. Generally, the memory is insufficient because the task needs to read a large number of small files.

Solution

Step 1 Check whether multiple HDFS files need to be read for the started MapReduce tasks. If yes, reduce the file quantity by combining the small-sized files in advance or using **combineInputFormat**.

Step 2 Increase the memory when the **hadoop** command is run. The memory is set on the client. Change the value of **-Xmx** in **CLIENT_GC_OPTS** in the *Client installation directory*/**HDFS/component_env** file to a larger value, for example, 512 MB. Run the **source component_env** command for the modification to take effect.

```
export YARN_ROOT_LOGGER=INFO,console

#GC_OPTS for client operation.
CLIENT_GC_OPTS="-Xmx512m -Djava.io.tmpdir=${HADOOP_HOME}"

export HADOOP_CLIENT_OPTS="$CLIENT_GC_OPTS"
```

----End

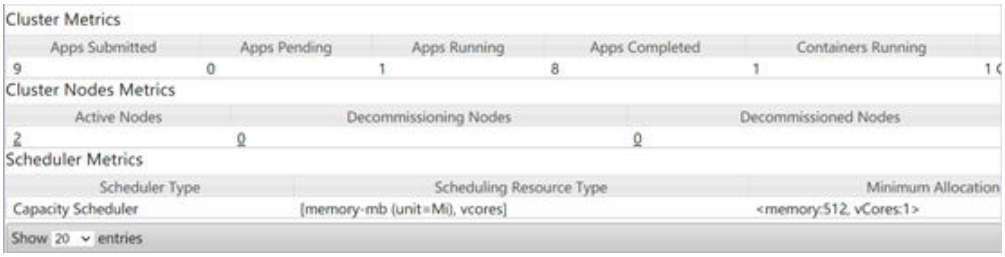
16.20.3 Disk Space Is Used Up Due to Oversized Aggregated Logs of Yarn

Issue

The disk usage of the cluster is high.

Symptom

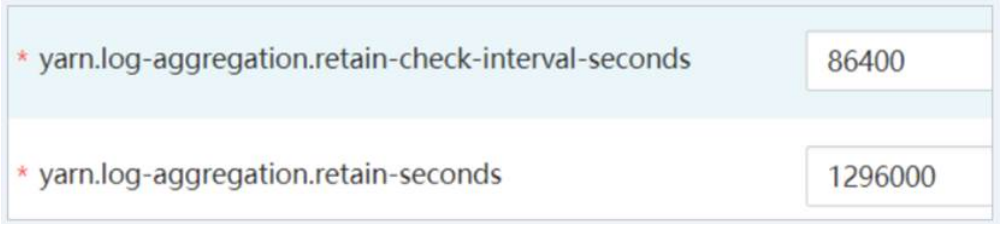
- On the host management page of Manager, the disk usage is too high.
- Only a few tasks are running on the Yarn web UI.



- After the **hdfs dfs -du -h /** command is executed on the master node of the cluster, the command output shows that the following files consume a large amount of disk space.

```
22.5 G   45.0 G   /tmp/logs/root/logs/application_1589278244866_0153
18.4 M   36.8 M   /tmp/logs/root/logs/application_1589278244866_0154
23.4 G   46.8 G   /tmp/logs/root/logs/application_1589278244866_0155
23.5 G   46.9 G   /tmp/logs/root/logs/application_1589278244866_0156
23.7 G   47.4 G   /tmp/logs/root/logs/application_1589278244866_0157
23.7 G   47.4 G   /tmp/logs/root/logs/application_1589278244866_0158
22.5 G   45.0 G   /tmp/logs/root/logs/application_1589278244866_0159
18.5 M   37.0 M   /tmp/logs/root/logs/application_1589278244866_0160
22.5 G   45.0 G   /tmp/logs/root/logs/application_1589278244866_0161
18.8 M   37.6 M   /tmp/logs/root/logs/application_1589278244866_0162
24.0 G   48.0 G   /tmp/logs/root/logs/application_1589278244866_0163
121.3 K  242.7 K  /tmp/logs/root/logs/application_1589278244866_0164
1.1 M    2.1 M    /tmp/logs/root/logs/application_1589278244866_0165
1.1 M    2.1 M    /tmp/logs/root/logs/application_1589278244866_0166
1.1 M    2.1 M    /tmp/logs/root/logs/application_1589278244866_0167
1.1 M    2.1 M    /tmp/logs/root/logs/application_1589278244866_0168
```

- The log aggregation configuration of the Yarn service is as follows.



Cause Analysis

Jobs are submitted too frequently, and the time for deleting aggregated log files is set to 1296000, that is, aggregated logs are retained for 15 days. As a result, aggregated logs cannot be released within a short period of time, exhausting the disk space.

Procedure

Step 1 Log in to Manager and navigate to the all configurations page of the MapReduce service.

- MRS Manager: Log in to MRS Manager, choose **Services** > **MapReduce** > **Service Configuration**, and select **All** from the **Type** drop-down list.
- FusionInsight Manager: Log in to FusionInsight Manager and choose **Cluster** > **Services** > **MapReduce**. On the MapReduce page, choose **Configurations** > **All Configurations**.

Step 2 Search for the **yarn.log-aggregation.retain-seconds** parameter and decrease its value based on site requirements, for example, to **259200**. In this case, the aggregated logs of Yarn are retained for three days, and the disk space is automatically released after the retention period expires.

Step 3 Click **Save Configuration** and deselect **Restart the affected services or instances**.

Step 4 Restart the MapReduce service during off-peak hours. The restart will interrupt upper-layer services and affect cluster management, maintenance, and services.

1. Log in to Manager.
2. Restart the MapReduce service.

----End

16.20.4 Temporary Files Are Not Deleted When an MR Job Is Abnormal

Issue

Temporary files are not deleted when an MR job is abnormal.

Symptom

There are too many files in the HDFS temporary directory, occupying too much memory.

Cause Analysis

When an MR job is submitted, related configuration files, JAR files, and files added by running the **-files** command are stored in the temporary directory on HDFS so that the started container can obtain the files. The configuration item **yarn.app.mapreduce.am.staging-dir** specifies the storage path. The default value is **/tmp/hadoop-yarn/staging**.

After a properly running MR job is complete, temporary files are deleted. However, when a Yarn task corresponding to the job exits abnormally, temporary files are not deleted. As a result, the number of files in the temporary directory increases over time, occupying more and more storage space.

Procedure

Step 1 Log in to a cluster.

1. Log in to any master node as user **root**. The user password is the one defined during cluster creation.
2. If Kerberos authentication is enabled for the cluster, run the following commands to go to the client installation directory and configure environment variables. Then, authenticate the user and enter the password as prompted. Obtain the password from an administrator.

   cd *Client installation directory*

   source bigdata_env

   kinit hdfs
3. If Kerberos authentication is not enabled for the cluster, run the following commands to switch to user **omm** and go to the client installation directory to configure environment variables:

   su - omm

   cd *Client installation directory*

   source bigdata_env

Step 2 Obtain the file list.

hdfs dfs -ls /tmp/hadoop-yarn/staging/*/.staging/ | grep "^drwx" | awk '{print $8}' > job_file_list

The **job_file_list** file contains the folder list of all jobs. The following shows an example of the file content:

```
/tmp/hadoop-yarn/staging/omm/.staging/job__<Timestamp>_<ID>
```

Step 3 Collect statistics on running jobs.

mapred job -list 2>/dev/null | grep job_ | awk '{print $1}' > run_job_list

The **run_job_list** file contains the IDs of running jobs. The content format is as follows:

```
job_<Timestamp>_<ID>
```

Step 4 Delete running jobs from the **job_file_list** file. Ensure that data of running jobs is not deleted by mistake when deleting expired data.

cat run_job_list | while read line; do sed -i "/$line/d" job_file_list; done

Step 5 Delete expired data.

cat job_file_list | while read line; do hdfs dfs -rm -r $line; done

Step 6 Delete temporary files.

rm -rf run_job_list job_file_list

----End

16.20.5 ResourceManager of Yarn (Port 8032) Throws Error "connection refused"

Issue

The ResourceManager of Yarn that requests to submit jobs throws error "connection refused", and the port number configured for Yarn is 8032.

Symptom

One of Yarn's ResourceManager nodes in the MRS cluster cannot be connected, and the port number configured for Yarn is 8032.

Cause Analysis

The service application runs outside the cluster, and the in-use client does not match the latest client configuration provided by the MRS cluster. The Yarn port is 8032, which is different from the actual port of Yarn's ResourceManager of MRS. As a result, the ResourceManager of Yarn that requests to submit jobs reports error "connection refused".

Procedure

Step 1 Update the MRS client.

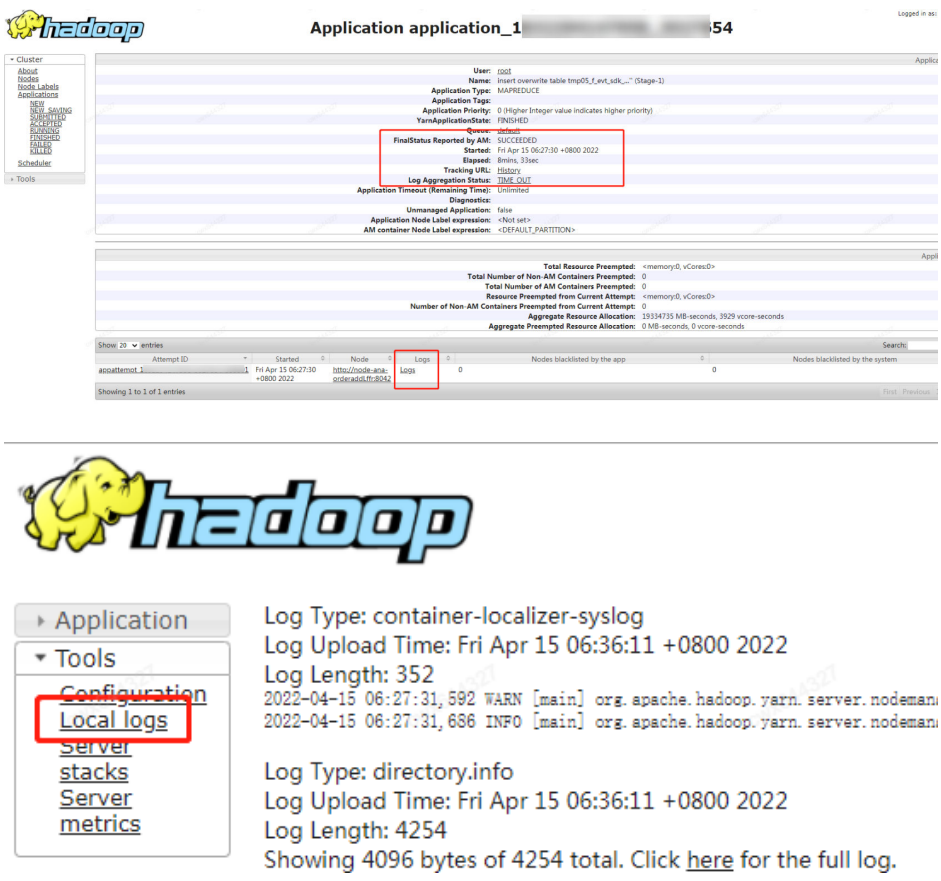
Step 2 Submit the job again.

----End

16.20.6 Failed to View Job Logs on the Yarn Web UI

Symptom

When a user logs in to the Yarn web UI to view job logs and clicks **Local logs**, error message "Could not access logs page!" is displayed.



Cause Analysis

Local logs is used to access service logs. However, for security purposes, this function is inaccessible from the Yarn web UI. You can log in to the active ResourceManager node to view ResourceManager logs.

Procedure

Step 1 Log in to Manager and choose **Cluster** > **Services** > **Yarn**. On the **Yarn** page, click the **Instance** tab and take note of the service IP address of the active ResourceManager instance.

Step 2 Log in to the active ResourceManager node as user **root**.

Step 3 Go to the **/var/log/Bigdata/yarn/rm** directory and view the ResourceManager logs.

cd /var/log/Bigdata/yarn/rm

----End

16.20.7 An Error Is Reported When a Queue Name Is Clicked on the Yarn Page

Symptom

When Yarn uses the Capacity scheduler, error 500 is reported after a user clicks a queue name on the native Yarn web UI.

HTTP ERROR 500 javax.servlet.ServletException: javax.servlet.ServletException: java.lang.IllegalArgumentException: Illegal character in query at index 81: https://[redacted]:20026/Yarn/ResourceManager/21/cluster/scheduler?openQueues=^default$

Cause Analysis

Symbol ^ in the URL cannot be identified. As a result, the page access fails.

Procedure

Step 1 Log in to Manager and choose **Cluster** > **Services** > **Yarn** > **Configurations** > **All Configurations**.

Step 2 Search for **yarn.resourcemanager.webapp.pagination.enable** in the search box.

Step 3 If the value is **true** (default), change it to **false** and save the configuration.

Step 4 On the Yarn page, click **Instance**, select all ResourceManager instances, click **More**, and select **Instance Rolling Restart**. Wait until the instances are started.

----End

16.21 Using ZooKeeper

16.21.1 Accessing ZooKeeper from an MRS Cluster

Issue

An error is reported when a user attempts to access ZooKeeper from an MRS cluster.

Symptom

The customer uses **zkcli.sh** to access ZooKeeper on the MRS Master node, but an error is reported.

Cause Analysis

The command used by the customer is incorrect. As a result, an error is reported.

Procedure

Step 1 Obtain the ZooKeeper IP address.

Step 2 Log in to the Master node as user **root**.

Step 3 Run the following command to initialize environment variables:

source /opt/client/bigdata_env

Step 4 Run the **zkCli.sh -server** *IP address of the node where ZooKeeper is located*:**2181** command to connect to ZooKeeper of the MRS cluster.

The IP address of the node where ZooKeeper is located is the one queried in **Step 1**. Use commas (,) to separate multiple IP addresses.

Step 5 Run common commands such as **ls /** to view ZooKeeper information.

----End

16.22 Accessing OBS

16.22.1 When Using the MRS Multi-user Access to OBS Function, a User Does Not Have the Permission to Access the /tmp Directory

Issue

When the MRS multi-user access to OBS function is used to execute jobs such as Spark, Hive, and Presto jobs, an error message is displayed, indicating that the user does not have the permission to access the **/tmp** directory.

Symptom

When the MRS multi-user access to OBS function is used to execute jobs such as Spark, Hive, and Presto jobs, an error message is displayed, indicating that the user does not have the permission to access the **/tmp** directory.

Cause Analysis

A temporary directory exists during job execution. The user who submits the job does not have permission on the temporary directory.

Procedure

Step 1 On the **Dashboard** tab page of the cluster, query and record the name of the agency bound to the cluster.

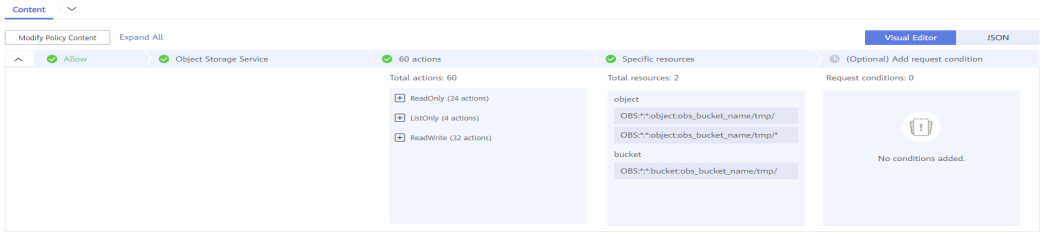
Step 2 Log in to the IAM console.

Step 3 Choose **Permissions**. On the displayed page, click **Create Custom Policy**.

- **Policy Name**: Enter a policy name.
- **Scope**: Select **Global services**.
- **Policy View**: Select **Visual editor**.
- **Policy Content**:
  a. **Allow**: Select **Allow**.
  b. **Select service**: Select **Object Storage Service (OBS)**.
  c. **Select action**: Select **WriteOnly**, **ReadOnly**, and **ListOnly**.
  d. **Specific resources**:
     i. Set **object** to **Specify resource path**, click **Add resource path**, and enter *obs_bucket_name*/**tmp/** and *obs_bucket_name*/**tmp/*** in **Path**. The **/tmp** directory is used as an example. If you need to add permissions for other directories, perform the following steps to add the directories and resource paths of all objects in the directories.
     ii. Set **bucket** to **Specify resource path**, click **Add resource path**, and enter *obs_bucket_name* in **Path**.

     Replace *obs_bucket-name* with the actual OBS bucket name. If the bucket type is Parallel File System, you need to add the *obs_bucket_name*/**tmp/** path. If the bucket type is Object Storage, you do not need to add the path.
  e. (Optional) Request condition, which does not need to be added currently.

Figure 16-61 Custom policy



Step 4 Click **OK**.

Step 5 Select **Agency** and click **Assign Permissions** in the **Operation** column of the agency queried in **Step 1**.

Step 6 Query and select the created policy in **Step 3**.

Step 7 Click **OK**.

----End

16.22.2 When the Hadoop Client Is Used to Delete Data from OBS, It Does Not Have the Permission for the .Trash Directory

Issue

When a user uses the Hadoop client to delete data from OBS, an error message is displayed indicating that the user does not have the permission on the **.Trash** directory.

Symptom

After the **hadoop fs -rm obs://<*obs_path*>** command is executed, the following error information is displayed:

```
exception [java.nio.file.AccessDeniedException: user/root/.Trash/Current/: getFileStatus on user/root/.Trash/Current/: status [403]
```

Cause Analysis

When deleting a file, Hadoop moves the file to the **.Trash** directory. If the user does not have the permission on the directory, error 403 is reported.

Procedure

Solution 1:

Run the **hadoop fs -rm -skipTrash** command to delete the file.

Solution 2:

Add the permission to access the **.Trash** directory to the agency corresponding to the cluster.

Step 1 On the **Dashboard** tab page of the cluster, query and record the name of the agency bound to the cluster.

Step 2 Log in to the IAM console.

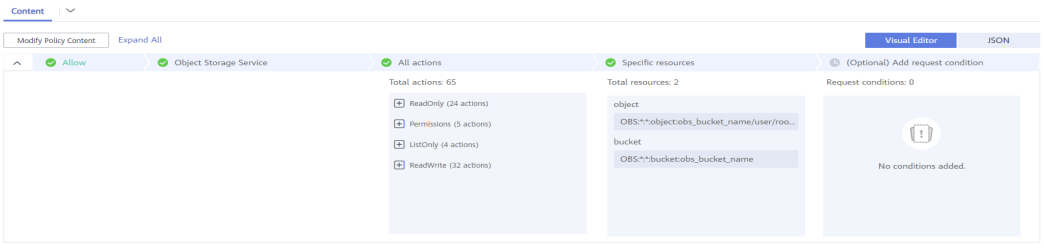
Step 3 Choose **Permissions**. On the displayed page, click **Create Custom Policy**.

- **Policy Name**: Enter a policy name.
- **Scope**: Select **Global services**.
- **Policy View**: Select **Visual editor**.
- **Policy Content**:
  a. **Allow**: Select **Allow**.
  b. **Select service**: Select **Object Storage Service (OBS)**.
  c. Select all operation permissions.
  d. **Specific resources**:
     i. Set **object** to **Specify resource path**, click **Add resource path**, and enter the **.Trash** directory, for example, **obs_bucket_name/user/root/.Trash/*** in **Path**.
     ii. Set **bucket** to **Specify resource path**, click **Add resource path**, and enter *obs_bucket_name* in **Path**.

Replace *obs_bucket-name* with the actual OBS bucket name.

e. (Optional) Request condition, which does not need to be added currently.

Figure 16-62 Custom policy



Step 4 Click **OK**.

Step 5 Select **Agency** and click **Assign Permissions** in the **Operation** column of the agency queried in **Step 1**.

Step 6 Query and select the created policy in **Step 3**.

Step 7 Click **OK**.

Step 8 Run the **hadoop fs -rm obs://<*obs_path*>** command again.

----End

17 Appendix

17.1 Precautions for MRS 3.x

Purpose

Custers of versions earlier than MRS 3.x use MRS Manager to manage and monitor MRS clusters. On the Cluster Management page of the MRS management console, you can view cluster details, manage nodes, components, alarms, patches, files, jobs, tenants, and backup and restoration. In addition, you can configure Bootstrap actions and manage tags.

MRS 3.x uses FusionInsight Manager to manage and monitor clusters. On the Cluster Management page of the MRS management console, you can view cluster details, manage nodes, components, alarms, files, jobs, Bootstrap actions, and tags.

Some maintenance operations of the MRS 3.x cluster are different from those of earlier versions. For details, see **MRS Manager Operation Guide (Applicable to 2.x and Earlier Versions)** and **FusionInsight Manager Operation Guide (Applicable to 3.x)**.

Accessing MRS Manager

- For details about how to access MRS Manager of versions earlier than MRS 3.x, see **Accessing MRS Manager MRS 2.1.0 or Earlier)**.
- For details about how to access FusionInsight Manager of MRS 3.x, see **Accessing FusionInsight Manager (MRS 3.x or Later)**.

Modifying MRS Cluster Service Configuration Parameters

- For versions earlier than MRS 3.x, you can modify service configuration parameters on the cluster management page of the MRS management console.
  a. Log in to the MRS console. In the left navigation pane, choose **Clusters** > **Active Clusters**, and click a cluster name.
  b. Choose **Components** > *Name of the desired service* > **Service Configuration**.

The **Basic Configurations** tab page is displayed by default. To modify more parameters, click the **All Configurations** tab. The navigation tree displays all configuration parameters of the service. The level-1 nodes in the navigation tree are service names or role names. The parameter category is displayed after the level-1 node is expanded.

c. In the navigation tree, select the specified parameter category and change the parameter values on the right.

If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The system searches for the parameter in real time and displays the result.

d. Click **Save Configuration**. In the displayed dialog box, click **OK**.

e. Wait until the message "Operation succeeded" is displayed. Click **Finish**. The configuration is modified.

Check whether there is any service whose configuration has expired in the cluster. If yes, restart the corresponding service or role instance for the configuration to take effect. You can also select **Restart the affected services or instances** when saving the configuration.

- In MRS 3.x, you need to log in to FusionInsight Manager to modify service configuration parameters.

a. Log in to FusionInsight Manager.

b. Choose **Cluster** > **Services**.

c. Click the specified service name on the service management page.

d. Click **Configurations**.

The **Basic Configurations** tab page is displayed by default. To modify more parameters, click the **All Configurations** tab. The navigation tree displays all configuration parameters of the service. The level-1 nodes in the navigation tree are service names or role names. The parameter category is displayed after the level-1 node is expanded.

e. In the navigation tree, select the specified parameter category and change the parameter values on the right.

If you are not sure about the location of a parameter, you can enter the parameter name in search box in the upper right corner. The Manager searches for the parameter in real time and displays the result.

f. Click **Save**. In the confirmation dialog box, click **OK**.

g. Wait until the message "Operation succeeded" is displayed. Click **Finish**. The configuration is modified.

Check whether there is any service whose configuration has expired in the cluster. If yes, restart the corresponding service or role instance for the configuration to take effect.